

HDF to GeoTIFF Science Processing Algorithm (H₂G_SPA) User's Guide

Version 2.3

October 2013



**GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND**

Table of Contents

1	General.....	1
2	Algorithm Wrapper Concept.....	1
3	Software Description.....	1
4	Software Version	2
5	Credits	3
6	Prerequisites.....	3
7	Program Inputs and Outputs.....	3
8	Installation and Configuration	3
8.1	Installing as a Standalone Application	3
8.2	Installing into an IPOPP Framework	4
9	Software Package Testing and Validation	5
10	Program Operation.....	6
10.1	To Use the Run Scripts	6
10.2	To Use the Scripts in the Testscripts Directory.....	10
11	H ₂ G Image Products	11
11.1	Standard Products.....	11
11.1.1	Generating H ₂ G Standard Products.....	18
11.1.2	Overriding Projection and Resolution of Standard Products	19
11.2	User-defined Products.....	21
11.2.1	Configuration Files for User-defined Products	21
11.2.2	Overriding Projection and Resolution for User-defined Products	30
11.3	Subsetting and Mosaicing	30
	Appendix A. Dataset Identifiers.....	A-1
	Appendix B. Geolocation Identifiers.....	B-1
	Appendix C. Modifying Maximum Java Heap Size.....	C-1
	Appendix D. H ₂ G Standard Product Descriptions	D-1

1 General

The NASA Goddard Space Flight Center's (GSFC) Direct Readout Laboratory (DRL), Code 606.3 developed this software for the International Polar Orbiter Processing Package (IPOPP). The IPOPP package maximizes the utility of Earth science data for making real-time decisions by giving fast access to instrument data and derivative products from the Suomi National Polar-orbiting Partnership (SNPP), Aqua, and Terra missions and, in the future, the Joint Polar Satellite System (JPSS) mission.

Users must agree to all terms and conditions in the Software Usage Agreement on the DRL Web Portal before downloading this software.

Software and documentation published on the DRL Web Portal may occasionally be updated or modified. The most current versions of DRL software are available at the DRL Web Portal:

<http://directreadout.sci.gsfc.nasa.gov/?id=software>

Questions relating to the contents or status of this software and its documentation should be addressed to the DRL via the Contact DRL mechanism at the DRL Web Portal:

<http://directreadout.sci.gsfc.nasa.gov/?id=dspContent&cid=66>

2 Algorithm Wrapper Concept

The DRL has developed an algorithm wrapper to provide a common command and execution interface to encapsulate multi-discipline, multi-mission science processing algorithms. The wrapper also provides a structured, standardized technique for packaging new or updated algorithms with minimal effort.

A Science Processing Algorithm (SPA) is defined as an algorithm to which the wrapper has been applied. SPAs will function in a standalone, cross-platform environment to serve the needs of the broad Direct Readout community. Detailed information about SPAs and other DRL technologies is available the DRL Web Portal.

3 Software Description

This DRL software package contains the H₂G_SPA (Hierarchical Data Format [HDF] to Georeferenced Tagged Image File Format [GeoTIFF] Converter Science Processing Algorithm). H₂G_SPA is specially designed for Direct Readout applications and can create geolocated GeoTIFF images, jpeg browse images, and png browse images for various parameter datasets in SNPP SPA products and MODIS Level 2 SPA products. H₂G also creates standard true color images and user-defined false color images from supported VIIRS and MODIS science products. The H₂G_SPA functions in two modes: Standalone, or as an IPOPP plug-in.

The geolocated GeoTIFF images are Geographic Information System (GIS)-ingestible and can also be opened by standard image viewers. The non-geolocated jpeg and png images are more suitable as browse images. These browse images are enhanced with vector overlays of land/sea and political boundaries.

H₂G_SPA incorporates the following features to enhance output images and facilitate scientific interpretation:

- while creating images from a primary dataset, a secondary dataset may be used to mask appropriate areas;
- user-defined color map and user-defined scaling capabilities for conversion of dataset values into image pixels;
- choice of either geographic or stereographic projection for the output image;
- optional subsetting of swaths into user-defined regions of interest;
- optional mosaicing of multiple swaths;
- jpeg and png browse images with legends;
- fire pixel overlays on other imagery (e.g., fire pixel overlays on TrueColor image).

H₂G currently allows the user to select geographic and stereographic projections. Inclusion of other projections is under consideration for future releases of H₂G_SPA.

4 Software Version

Version 1.2 of the DRL algorithm wrapper was used to package the SPA described in this document. The SPA uses H₂G processing code (Version 2.3, September 2013). The H₂G_SPA stereographic projection capability utilizes the JPROJ.4 Java Native Interface (JNI) to the PROJ.4 Cartographic Projections Library. This library was initially developed by the U.S. Geological Survey (USGS) and is currently being maintained/enhanced by the Open Source Geospatial Foundation (OSGeo).

Enhancements to H₂G Version 2.3 include:

- addition of documentation and test script support for SNPP VIIRS EDR products (please refer to Table 2);
- extended support for VIIRS Ocean Color and SST science products;
- extended support for OMPS SO₂, Aerosol Index, Ozone and Reflectivity science products;
- ability to generate imagery from Daily composite products (like the Level 2G products produced by the MODIS and VIIRS Burnscar SPAs);
- ability to generate a high resolution VIIRS True Color image product using the VIIRS I1 Band;

- ability to support global mosaiced image products.

This software will execute on a 64-bit computer, and has been tested on computers with 32 GB RAM with the following operating systems:

- a) Fedora 18 X86_64;
- b) CentOS Linux 6.4 X86_64;
- c) OpenSUSE Linux 12.1 X86_64;
- d) Kubuntu 12.04 X86_64.

Copyright 1999-2007, United States Government as represented by the Administrator for the National Aeronautics and Space Administration. All Rights Reserved.

5 Credits

H₂G was developed by the DRL at NASA/GSFC.

6 Prerequisites

To run this package, you must have the Java Development Kit (JDK) or Java Runtime Engine (JRE) (Java 1.6.0_25 or higher) installed on your computer, and the bin directory of your Java installation in your PATH environment variable.

7 Program Inputs and Outputs

H₂G_SPA supports various Level 2 MODIS, VIIRS SDR, VIIRS EDR/IP SPA and OMPS products as input. H₂G_SPA has been pre-configured to produce a number of standard image products. However, users can configure H₂G_SPA to produce non-standard image products.

8 Installation and Configuration

H₂G_SPA can be run either as a standalone application or can be installed dynamically into an IPOPP framework as a plug-in.

8.1 Installing as a Standalone Application

Download the H2G_2.3_SPA_1.2.tar.gz and H2G_2.3_SPA_1.2_testdata.tar.gz (optional) files into the same directory.

Decompress and un-archive the H2G_2.3_SPA_1.2.tar.gz and H2G_2.3_SPA_1.2_testdata.tar.gz (optional) files:

```
$ tar -xzf H2G_2.3_SPA_1.2.tar.gz
$ tar -xzf H2G_2.3_SPA_1.2_testdata.tar.gz
```

This will create the following subdirectories:

```
SPA
  h2g
    algorithm
    station
    wrapper
    ancillary
    testscripts
    testdata
```

H2G_SPA was compiled with Java 1.6. H2G_SPA is configured to run with a maximum Java heap size of 2GB. H2G_SPA can fail on a computer with inadequate memory. Users may increase or decrease the memory specifications. Refer to Appendix C for instructions.

NOTE: Examples supplied from this point forward assume that the SPA was installed into /home/ipopp/drl.

8.2 Installing into an IPOPP Framework

H2G_SPA can be installed dynamically into an IPOPP framework. The IPOPP Control Services (CS) stations corresponding to the standard H2G products have been listed in Table 2. Instructions for installing this SPA into IPOPP and enabling the corresponding CS stations are contained in the IPOPP User's Guide (available on the DRL web portal).

Pre-configuring H2G for stereographic imagery: H2G_SPA is pre-configured to produce geographically projected GeoTIFFs in IPOPP Mode. Users can configure H2G_SPA in IPOPP Mode to produce stereographically projected GeoTIFFs instead by performing an optional configuration step prior to issuing the NISGSinstall.sh command, as follows:

cd into the algorithm directory and execute the configure-ipopp-projection script. Input '2' when asked for the projection options for IPOPP Mode, as follows:

```
$ cd /home/ipopp/drl/SPA/h2g/algorithm
$ ./configure-ipopp-projection
```

Projection Options for IPOPP Mode: Select One

1. Geographic 2. Stereographic

2

Configuring h2g stations for Stereographic projections

H₂G_SPA is now ready to be installed into the IPOPP framework as described in the IPOPP User's Guide.

Pre-configuring H₂G for generating daily composite imagery: H₂G can generate daily composited true-color, false color and firemask imagery in IPOPP mode. These image products are generated using daily composite tiled products available from the MODIS BURNSCAR_SPA. H₂G requires a couple of pre-configuration steps to enable this capability before installation into IPOPP.

1. Identify the MODIS tile(s) that has been enabled in the MODIS BURNSCAR_SPA (See BURNSCAR_SPA User's Guide for details). cd to the SPA/h2g/algorithm/tileStations/ directory and edit the ipopp_TileList.cfg file. This file includes a list of tileIDs on successive lines. Please add the same tileIDs that have been enabled in BURNSCAR_SPA.
2. Now execute the configure-burnscar script. Input '1' or '2' depending on what projection you want for the daily composite images.

```
$. ./configure-burnscar
Projection Options for IPOPP Mode: Select One
1. Geographic 2. Stereographic
1
Configuring h2g stations for Geographic projections
Tiles selected for processing: hxxvyy ... ..
Configuring tcolor-tile station tile1 : hxxvyy
...
...
```

H₂G_SPA is now ready to be installed into the IPOPP framework as described in the IPOPP User's Guide.

NOTE: The configure-ipopp-projection and the configure-burnscar scripts can be re-executed to switch to a different projection in IPOPP Mode. However, once reconfiguration is complete, the user must reinstall H₂G_SPA into the IPOPP Framework as instructed in the IPOPP User's Guide.

9 Software Package Testing and Validation

The testscripts subdirectory contains test scripts that can be used to verify that your current installation of the SPA is working properly, as described below. Note that the optional H2G_2.3_SPA_1.2_testdata.tar.gz file is required to execute these testing procedures.

Step 1: cd into the testscripts directory.

Step 2: Run the 'run-lst-tiff' script by typing:

```
$/run-lst-tiff
```

A successful execution usually takes some time, so if the execution seems to get stuck, do not become impatient. If everything is working properly, the script will terminate with a message such as:

```
Output h2gout is /home/ipopp/drl/SPA/h2g/testdata/output/LST.tif
```

You can cd to the output directory to verify that the science product (in this example the Land Surface Temperature [LST] GeoTIFF image) exists. If it does exist, then the process executed successfully. If there is a problem and the code terminates abnormally, the problem can be identified using the log files. Log files are automatically generated within the directory used for execution. They start with `stdfile*` and `errfile*`. Other log and intermediate files may be generated automatically within the directory used for execution. They are useful for traceability and debugging purposes. However it is strongly recommended that users clean up log files and intermediate files left behind in the run directory before initiating a fresh execution of the SPA. Intermediate files from a previous run may affect a successive run and produce ambiguous results. Please report any errors that cannot be fixed to the DRL. Test output file(s) are provided for comparison in the `testdata/output/h2g_standard_outputs` directory. The output products serve as an indicator of expected program output. Use a comparison utility (such as a standard image viewer) to compare your output(s) to those provided in the `testdata/output/h2g_standard_outputs` subdirectory.

10 Program Operation

In order to run the package using your own input data, you can either use the run scripts within the wrapper directory, or modify the test scripts within the testscripts directory.

10.1 To Use the Run Scripts

Identify the 'run' script: The wrapper directory within this package contains the `h2g` subdirectory. You must execute the 'run' within the `h2g/wrapper/h2g` subdirectory to execute `H2G_SPA`. Note that to execute 'run', you need to have java on your path.

Specify input parameters using <label value> pairs: To execute the 'run' script, you must supply the required input and output parameters. Input and output parameters are usually file paths or other values (e.g., the output type). Each parameter is specified on the command line by a <label value> pair. Labels are

simply predefined names for parameters. Each label must be followed by its actual value. The <label value> pairs must be specified on the command line in order for H₂G_SPA to execute. Some of these pairs are optional, meaning the process would still be able to execute even if that parameter is not supplied. There are three types of <label value> pairs that the H₂G_SPA uses, as follows:

- a) Input file label/values. These are input file paths. Values are absolute or relative paths to the corresponding input file.
- b) Parameter label/values. These are parameters that need to be passed onto the SPA (e.g., the image output type).
- c) Output file labels. These are output files that are produced by the SPA. Values are the relative/absolute paths of the files you want to generate.

The following table lists and describes the labels associated with the SPA. Section 11, "H₂G Image Products," contains detailed descriptions and examples of usage of these labels.

Table 1. Labels and Descriptions Required by H₂G

Input File Labels	Description
input.data	Path to the supported sensor data product (HDF4/HDF5 file) that contains the parameter dataset for which the image is being created.
input.data n (n=2,3..10)	Path to the n th supported sensor data product (HDF4/HDF5 file) that contains the parameter dataset for which a mosaiced image is being created.
red.data	Path to the supported sensor data product (HDF4/HDF5 file) that contains the dataset which would be used as the red band during rgb image generation. This label is used along with green.data and blue.data labels when the red, green and blue bands are in different HDF files.
red.data n (n=2,3..10)	Path to the n th supported sensor data product (HDF4/HDF5 file) that contains the dataset which would be used as the red band during a mosaiced rgb image generation. This label is used along with the corresponding n th green.data n and blue.data n labels when the red, green and blue bands are in different HDF files.
green.data	Path to the supported sensor data product (HDF4/HDF5 file) that contains the dataset which would be used as the green band during rgb image generation. This label is used along with red.data and blue.data labels when the red, green and blue bands are in different HDF files.
green.data n (n=2,3..10)	Path to the n th supported sensor data product (HDF4/HDF5 file) that contains the dataset which would be used as the green band during a mosaiced rgb image generation. This label is used along with the corresponding n th red.data n and blue.data n labels when the red, green and blue bands are in different HDF files.
blue.data	Path to the supported sensor data product (HDF4/HDF5 file) that contains the dataset which would be used as the blue band during rgb image generation. This label is used along with green.data and red.data labels when the red, green and blue bands are in different HDF files.
blue.data n (n=2,3..10)	Path to the n th supported sensor data product (HDF4/HDF5 file) that contains the dataset which would be used as the blue band during a mosaiced rgb image generation. This label is used along with the corresponding n th green.data n and red.data n labels when the red, green and blue bands are in different HDF files.
mask (optional)	Path to the supported sensor data product (HDF4/HDF5 file) that contains the mask dataset used as a mask for the image being created. mask is not needed if either no mask is being used or the mask dataset is contained within input.data.
mask n (n=2,3..10)	Path to the n th supported sensor data product (HDF4/HDF5 file)

	that contains the mask dataset used as a mask for the mosaiced image being created. <i>maskn</i> is not needed if either no mask is being used or the mask dataset is contained within <i>input.data</i> .
geo (optional)	Path to the supported sensor geolocation data product (HDF4/HDF5 file) or the HDF file which contains the latitude and longitude datasets for the same swath. <i>geo</i> is not needed only when the geolocation information is within the <i>input.data</i> file (e.g., HDF outputs from IMAPP_SPA have their own geolocation).
geon (n=2,3..10)	Path to the <i>n</i> th supported sensor geolocation data product (HDF4/HDF5 file) or the HDF file which contains the latitude and longitude datasets for the <i>n</i> th swath product during mosaicing. <i>geon</i> is not needed only when the geolocation information is within the <i>input.data</i> file (e.g., HDF outputs from IMAPP_SPA have their own geolocation).
fireloc (optional)	Path to the fire-location text file. The fire-location text file is produced by the MOD14_SPA. <i>fireloc</i> can be used to overlay fire pixels on the output image. <i>fireloc</i> can be used only when the image output type is <i>geotiff.rgb</i> , <i>jpeg.rgb</i> or <i>png.rgb</i> . (See <i>output.type</i> label description below.) The configuration file in paragraph 11.2.1.2 must contain the FIRESOURCE keyword for fire pixel overlays to work.
firelocn (n=2,3..10)	Path to the <i>n</i> th fire-location text file. The fire-location text file is produced by the MOD14_SPA. <i>firelocn</i> can be used to overlay fire pixels on the mosaiced output image. <i>firelocn</i> can be used only when the image output type is <i>geotiff.rgb</i> , <i>jpeg.rgb</i> or <i>png.rgb</i> . (See <i>output.type</i> label description below.) The configuration file (described in paragraph 11.2.1.2) must contain the FIRESOURCE keyword for fire pixel overlays to work.
Output File Label	Description
h2gout	Path to the geotiff, jpeg or png image product generated by H ₂ G.
Parameter Label	Description
config.type	Configuration type. Can be (a) 'standard' for the standard H ₂ G products; (b) 'singleband' for a user-defined single band image; or (c) 'rgb' for a user-defined RGB image.
config.name	<i>config.name</i> is either (a) the identifier for the standard H ₂ G products; or (b) the path to the user-defined configuration file.
output.type	<i>output.type</i> can be either (a) <i>geotiff.u8cm</i> (for an 8-bit colormap embedded GeoTIFF image); (b) <i>geotiff.rgb</i> (for an RGB GeoTIFF image); (c) <i>jpeg.rgb</i> (for an RGB jpeg image); or (d) <i>png.rgb</i> (for an RGB png image). NOTE: <i>jpeg.rgb</i> and <i>png.rgb</i> images do not have geolocation information. They are more useful as browse images. <i>jpeg.rgb</i> and <i>png.rgb</i> images have vector overlays of land/sea/political

	boundaries and may have legends.
projection (optional)	Projection can be either (a) geographic or (b) stereographic. This parameter is used to override the projection defined for the H ₂ G standard product or in the user-defined configuration file.
resolution (optional)	Used to specify the resolution of the output image product. When using the geographic projection, resolution is in degrees. For the stereographic projection, resolution is in meters.
Subsetting and Mosaicing parameters:	
centerlat centerlon (in degrees)	These two parameters are used for subsetting/mosaicing and represent the center of the output image product. For a stereographically projected image, this point also represents the center of projection.
height_km, width_km (in km, used for stereographic projections)	These two parameters are used during subsetting/mosaicing of stereographically projected image products. They represent the height and width of the output image in km. Together with centerlat and centerlon they describe the geographic extent of the output image.
height_lat, width_lon (in degrees, used for geographic projections)	These two parameters are used during subsetting/mosaicing of geographically projected image products. They represent the height and width of the output image in geographic degrees. Together with centerlat and centerlon they describe the geographic extent of the output image.

NOTE: The input.data, geo, mask HDF and fireloc text files must correspond to the same swath. Similarly the nth input.data_n, geo_n and mask_n HDF and fireloc_n text files must correspond to the same swath.

10.2 To Use the Scripts in the Testscripts Directory

One simple way to run the algorithms from any directory of your choice, using your own data, is to copy the corresponding run-xxx scripts from the testscripts directory and its subdirectories (more_examples and other_examples) to your selected directory. Change the values of the different variables to reflect the file paths of the wrapper directories and the input/output files. Then modify the input/output file name variables. If required, add more parameters to the command line. Run the scripts to process your data.

11 H₂G Image Products

This section describes:

- *What are H₂G standard products?*
 - *How to generate the standard products?*
 - *How to override the projection and resolution of the standard products?*
- *What are H₂G user-defined products?*
 - *How to write configuration files for user-defined products?*
 - *How to override the projection and resolution defined in the configuration file?*
- *How to create subsetted and mosaiced image products.*

H₂G can produce two types of products: standard and user-defined. H₂G is pre-configured to generate the standard image products contained in Table 2, as described below in "Standard Products." Generation of user-defined (i.e., non-standard) image products, however, requires the user to undertake the additional step of writing a unique configuration file. These configuration files are simple text files that supply the user configuration to H₂G. Detailed instructions to generate standard and user-defined image products follow. Users are reminded that numerous examples of command-lines/configuration-files are contained within the H2G_2.3_SPA_1.2_testdata.tar.gz archive.

11.1 Standard Products

The standard outputs can be generated by setting the config.type label to 'standard' and using the correct identifier for the config.name label. Table 2 contains a list of standard outputs, with corresponding config.name identifiers and required input parameters. H₂G standard outputs are by default in geographic projection; instructions to override the default setting are contained in paragraph 11.1.2.

Table 2. H₂G Standard Outputs and Corresponding Input Requirements

'config.name' Identifiers for H2G Standard Products	Parameter (default resolution)	Inputs	H2G Stations Producing these Products (for IPOPP Mode)
ndvi	NDVI (masks: water and cloud; Resolution: 0.01 ⁰)	input.data <mod13 HDF output from NDVIEVI_SPA> mask <L2 HDF output from MOD14_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name ndvi output.type <geotiff.u8cm jpeg.argb png.argb>	ndvi-geotiff
evi	EVI (masks: water and cloud; Resolution: 0.01 ⁰)	input.data <mod13 HDF output from NDVIEVI_SPA> mask <L2 HDF output from MOD14_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name evi output.type <geotiff.u8cm jpeg.argb png.argb >	evi-geotiff
lst	LST (masks: water and cloud; Resolution: 0.01 ⁰)	input.data <LST HDF output from MODLST_SPA> mask <L2 HDF output from MOD14_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name lst output.type <geotiff.u8cm jpeg.argb png.argb>	lst-geotiff
fire	Fire Mask (Resolution: 0.01)	input.data <MOD14 HDF output from MOD14_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> fireloc <firelocation text file output from MOD14_SPA; ignored when output.type is geotiff.u8cm> config.type standard config.name fire output.type <geotiff.u8cm jpeg.argb png.argb >	fire-geotiff
sst	Sea Surface Temperature (Mask: SST quality flag; Resolution: 0.01 ⁰)	input.data <SST HDF output from L2GEN_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name sst output.type <geotiff.u8cm jpeg.argb png.argb >	sst-geotiff
chlor	Chlorophyll-a concentration (Mask: l2flags; Resolution: 0.01 ⁰)	input.data <Chlor HDF output from L2GEN_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name chlor output.type <geotiff.u8cm jpeg.argb png.argb >	chlor_a-geotiff
aerosol-aod	Aerosol Optical Depth (Resolution: 0.1 ⁰)	input.data <mod04 HDF output from IMAPP_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name aerosol-aod output.type <geotiff.u8cm jpeg.argb png.argb >	aerosols-geotiff
cloudtop-irphase	Cloud Phase (Resolution: 0.05 ⁰)	input.data <mod06 HDF output from IMAPP_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name cloudtop-irphase output.type <geotiff.u8cm jpeg.argb png.argb >	irphase-geotiff
cloudtop-ctp	Cloudtop Pressure (Resolution: 0.05 ⁰)	input.data <mod06 HDF output from IMAPP_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name cloudtop-ctp	ctp-geotiff

		output.type <geotiff.u8cm jpeg.rgb png.rgb>	
atmprofile-tpw	Total Precipitable Water (Resolution: 0.05 ⁰)	input.data <mod07 HDF output from IMAPP_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name atmprofile-tpw output.type <geotiff.u8cm jpeg.rgb png.rgb>	atmprofile-geotiff
cloudmask	Cloudmask (Resolution: 0.01 ⁰)	input.data <mod35 HDF output from IMAPP_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name cloudmask output.type <geotiff.u8cm jpeg.rgb png.rgb >	cloudmask-geotiff
tcolor0_0025	True Color from corrected reflectances (Resolution: 0.0025 ⁰)	input.data <crefl HDF output from CREFL_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolor0_0025 output.type <geotiff.rgb jpeg.rgb png.rgb >	N/A
tcolor0_005	True Color from corrected reflectances (Resolution: 0.005 ⁰)	input.data <crefl HDF output from CREFL_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolor0_005 output.type <geotiff.rgb jpeg.rgb png.rgb >	N/A
tcolor0_01	True Color from corrected reflectances (Resolution: 0.01 ⁰)	input.data <crefl HDF output from CREFL_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolor0_01 output.type <geotiff.rgb jpeg.rgb png.rgb >	creflrgb-geotiff
tcolorfire0_0025	True Color with fire pixel overlays from corrected reflectances (Resolution: 0.0025 ⁰)	input.data <crefl HDF output from CREFL_SPA> fireloc <firelocation text file output from MOD14_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolorfire0_0025 output.type <geotiff.rgb jpeg.rgb png.rgb>	N/A
tcolorfire0_005	True Color with fire pixel overlays from corrected reflectances (Resolution: 0.005 ⁰)	input.data <crefl HDF output from CREFL_SPA> fireloc <firelocation text file output from MOD14_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolorfire0_005 output.type <geotiff.rgb jpeg.rgb png.rgb>	N/A
tcolorfire0_01	True Color with fire pixel overlays from corrected reflectances (Resolution: 0.01 ⁰)	input.data <crefl HDF output from CREFL_SPA> fireloc <firelocation text file output from MOD14_SPA> geo <MOD03 HDF output from MODISL1DB_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolorfire0_01 output.type <geotiff.rgb jpeg.rgb png.rgb>	creflrgbfire-geotiff
mod09tcolor0_005	True Color from surface reflectances (Resolution: 0.005 ⁰)	input.data <mod09 HDF output from MOD09_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name mod09tcolor0_005 output.type <geotiff.rgb jpeg.rgb png.rgb>	N/A
vtcolor	VIIRS Top of Atmosphere True Color (Resolution 0.01 ⁰)	red.data <VIIRS SVM05 HDF > green.data <VIIRS SVM04 HDF > blue.data <VIIRS SVM03 HDF > geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vtcolor output.type <geotiff.rgb jpeg.rgb png.rgb>	vtoatcolor-geotiff
vdbnnight	VIIRS Day/Night Band (configured for nighttime images;	input.data <VIIRS SVDNB HDF > geo <VIIRS GDNBO HDF > h2gout <Path to tif, png or jpeg output image file>	vdbnnight-geotiff

	Resolution: 0.01 ⁰)	config.type standard config.name vdnbnight output.type <geotiff.rgb jpeg.rgb png.rgb>	
vdnbdlay	VIIRS Day/Night Band (configured for daytime images; Resolution: 0.01 ⁰)	input.data <VIIRS SVDNB HDF > geo <VIIRS GDNBO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vdnbdlay output.type <geotiff.rgb jpeg.rgb png.rgb>	vdnbdlay-geotiff
vm12bt	VIIRS M12 Brightness Temperature (Resolution: 0.01 ⁰)	input.data <VIIRS SVM12 HDF > geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vm12bt output.type <geotiff.rgb jpeg.rgb png.rgb>	vm12h5-geotiff
vcviirs	VIIRS True Color from CVIIRS Corrected Reflectances (Resolution: 0.01 ⁰)	input.data <VIIRS CVIIRS MOD output> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vcviirs output.type <geotiff.rgb jpeg.rgb png.rgb>	vcviirs-geotiff
vcviirsi	VIIRS Imagery Resolution True Color from CVIIRS Corrected Reflectances (Resolution: 0.005 ⁰)	red.data < CVIIRS IMG output> green.data <CVIIRS MOD output> blue.data <CVIIRS MOD output> geo <VIIRS GITCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vcviirsi output.type <geotiff.rgb jpeg.rgb png.rgb>	N/A
vimgmfcolor	VIIRS False Color from GTMImagery EDR M1, M4, M9 Band Reflectances (Not projected)	red.data <VIIRS VM010 HDF > green.data <VIIRS VM020 HDF > blue.data <VIIRS VM030 HDF > geo <VIIRS GMGTO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vimgmfcolor output.type <geotiff.rgb jpeg.rgb png.rgb>	vimgmfcolor-geotiff
vimgifcolor	VIIRS False Color from GTMImagery EDR I1, I2, I3 Band Reflectances (Not projected)	red.data <VIIRS VI3BO HDF > green.data <VIIRS VI2BO HDF > blue.data <VIIRS VI1BO HDF > geo <VIIRS GIGTO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vimgifcolor output.type <geotiff.rgb jpeg.rgb png.rgb>	vimgifcolor-geotiff
vnccalbedo	VIIRS NCC Albedo Color from GTMImagery NCC EDR (Not projected)	input.data <VIIRS VNNCO HDF > geo <VIIRS GNCCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vnccalbedo output.type <geotiff.rgb jpeg.rgb png.rgb>	vimgncc-geotiff
viirsaf	VIIRS Fire Mask (Resolution: 0.01 ⁰)	input.data <VIIRS VAF HDF output from VIIRSAF_SPA> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name viirsaf output.type <geotiff.rgb jpeg.rgb png.rgb>	viirsaf-geotiff
vtcolorfire	VIIRS True Color with fire pixel overlays (Resolution: 0.01 ⁰)	input.data <VIIRS CVIIRS MOD output> geo <VIIRS GMTCO HDF > fireloc <VIIRS VAF txt output from VIIRSAF_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vtcolorfire	vcviirsfire-geotiff

		output.type <geotiff.argb jpeg.argb png.argb>	
vmfh5	VIIRS AVAFO Fire Mask (Resolution 0.01 ⁰)	input.data <VIIRS AVAFO HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vmfh5 output.type <geotiff.argb jpeg.argb png.argb>	vmfh5-geotiff
vcmmaskh5	VIIRS Cloud Mask (Resolution 0.01 ⁰)	input.data <VIIRS IICMO HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vcmmaskh5 output.type <geotiff.argb jpeg.argb png.argb>	vcmmaskh5-geotiff
vcmphaseh5	VIIRS Cloud Phase (Resolution 0.01 ⁰)	input.data <VIIRS IICMO HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vcmphaseh5 output.type <geotiff.argb jpeg.argb png.argb>	vcmphaseh5-geotiff
vaot550h5d	VIIRS Aerosol Optical Thickness at 550nm (Resolution 0.01 ⁰)	input.data <VIIRS IVAOT HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vaot550h5d output.type <geotiff.argb jpeg.argb png.argb>	vaot550h5d-geotiff
vapsh5d	VIIRS Aerosol Particle Size (Resolution 0.01 ⁰)	input.data <VIIRS IVAOT HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vapsh5d output.type <geotiff.argb jpeg.argb png.argb>	vapsh5d-geotiff
vsumh5	VIIRS Suspended Matter Type (Resolution 0.01 ⁰)	input.data <VIIRS VSUMO HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vsumh5 output.type <geotiff.argb jpeg.argb png.argb>	vsumh5-geotiff
vccth5d	VIIRS Cloud Top Temperature (Resolution 0.01 ⁰)	input.data <VIIRS IVIWT HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vccth5d output.type <geotiff.argb jpeg.argb png.argb>	vccth5d-geotiff
vcoth5d	VIIRS Cloud Optical Thickness (Resolution 0.01 ⁰)	input.data <VIIRS IVCOP HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vcoth5d output.type <geotiff.argb jpeg.argb png.argb>	vcoth5d-geotiff
vepsh5d	VIIRS Cloud Effective Particle Size (Resolution 0.01 ⁰)	input.data <VIIRS IVCOP HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vepsh5d output.type <geotiff.argb jpeg.argb png.argb>	vepsh5d-geotiff
vsnowbinh5	VIIRS Snow Cover Binary Map (Resolution 0.01 ⁰)	input.data <VIIRS VSCMO HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file>	vsnowbinh5-geotiff

		file> config.type standard config.name vsnowbinh5 output.type <geotiff.rgb jpeg.rgb png.rgb>	
vsreflh5d	VIIRS Land Surface Reflectance True Color (Resolution 0.01°)	input.data <VIIRS IVISR HDF> geo <VIIRS GMTCO HDF > h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vsreflh5d output.type <geotiff.rgb jpeg.rgb png.rgb>	vsurfreflh5d-geotiff
vndvih5	VIIRS NDVI (Resolution 0.01°)	input.data <VIIRS VIVIO HDF> geo <VIIRS GMTCO HDF > mask <VIIRS AVAFO HDF> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vndvih5 output.type <geotiff.rgb jpeg.rgb png.rgb>	vndvih5-geotiff
vevih5	VIIRS EVI (Resolution 0.01°)	input.data <VIIRS VIVIO HDF> geo <VIIRS GMTCO HDF > mask <VIIRS AVAFO HDF> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vevih5 output.type <geotiff.rgb jpeg.rgb png.rgb>	vevih5-geotiff
vlsth5	VIIRS Land Surface Temperature (Resolution 0.01°)	input.data <VIIRS VLSTO HDF> geo <VIIRS GMTCO HDF > mask <VIIRS AVAFO HDF> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name vlsth5 output.type <geotiff.rgb jpeg.rgb png.rgb>	vlsth5-geotiff
viirsst	VIIRS Sea Surface Temperature (from l2gen_SPA) (Resolution 0.01°)	input.data <VIIRS SST HDF output (from L2GEN_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name viirsst output.type <geotiff.rgb jpeg.rgb png.rgb>	viirsst-geotiff
viirschlor	VIIRS Chlorophyll Concentration (from l2gen) (Resolution 0.01°)	input.data <VIIRS Ocean Color HDF output (from L2GEN_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name viirschlor output.type <geotiff.rgb jpeg.rgb png.rgb>	viirschlor-geotiff
viirsndvi	VIIRS NDVI (from VIIRS-VI_SPA) (Resolution 0.01°)	input.data <VIIRS VI HDF output (from VIIRS-VI_SPA)> geo <VIIRS GMTCO HDF > mask <VIIRS VAF HDF from VIIRS-AF_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name viirsndvi output.type <geotiff.rgb jpeg.rgb png.rgb>	viirsndvi-geotiff
viirsevi	VIIRS EVI (from VIIRS-VI_SPA) (Resolution 0.01°)	input.data <VIIRS VI HDF output (from VIIRS-VI_SPA)> geo <VIIRS GMTCO HDF > mask <VIIRS VAF HDF from VIIRS-AF_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name viirsevi output.type <geotiff.rgb jpeg.rgb png.rgb>	viirsevi-geotiff
viirslst	VIIRS LST (from	input.data <VIIRS LST HDF output (from VIIRS-	viirslst-geotiff

	VIIRS-LST_SPA) (Resolution 0.01 ⁰)	LST_SPA)> geo <VIIRS GMTCO HDF > mask <VIIRS VAF HDF from VIIRS-AF_SPA> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name viirslst output.type <geotiff.argb jpeg.argb png.argb>	
firetile	MODIS Active Fires Daily Composite (Resolution 0.01 ⁰)	input.data < Level3 Mod14 Daily composite Tiled product (output from BURNSCAR_SPA)> geo <Level 3 Daily Composite Tiled Geolocation (output from BURNSCAR_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name firetile output.type <geotiff.argb jpeg.argb png.argb>	mod14_tile_hxxvyy (Please replace 'xx' and 'yy' to represent tile selections made during H2G pre-configuration for daily composite imagery; See section 8.2)
tcolortile0_01	MODIS True Color Daily Composite (Resolution 0.01 ⁰)	input.data < Level3 Corrected Reflectance Daily composite Tiled product (output from BURNSCAR_SPA)> geo <Level 3 Daily Composite Tiled Geolocation (output from BURNSCAR_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name tcolortile0_01 output.type <geotiff.argb jpeg.argb png.argb>	tcolor_tile_hxxvyy (Please replace 'xx' and 'yy' to represent tile selections made during H2G pre-configuration for daily composite imagery; See section 8.2)
fcolortile0_01	MODIS False Color (Using MODIS bands 2, 5 and 7) Daily Composite (Resolution 0.01 ⁰)	input.data < Level3 Corrected Reflectance Daily composite Tiled product (output from BURNSCAR_SPA)> geo <Level 3 Daily Composite Tiled Geolocation (output from BURNSCAR_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name fcolortile0_01 output.type <geotiff.argb jpeg.argb png.argb>	fcolor_tile_hxxvyy (Please replace 'xx' and 'yy' to represent tile selections made during H2G pre-configuration for daily composite imagery; See section 8.2)
uvaerosol	OMPS Ultra Violet Aerosol (Resolution 0.1 ⁰)	input.data <OMPS Total Column Total Ozone HDF output (from OMPSnadir_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name uvaerosol output.type <geotiff.argb jpeg.argb png.argb>	ompsaot-geotiff
totalozone	OMPS Total Column Ozone (Resolution 0.1 ⁰)	input.data <OMPS Total Column Total Ozone HDF output (from OMPSnadir_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name totalozone output.type <geotiff.argb jpeg.argb png.argb>	ompsstozone-geotiff
uvrefl331	OMPS Ultra Violet Reflectance at 331nm (Resolution 0.1 ⁰)	input.data <OMPS Total Column Total Ozone HDF output (from OMPSnadir_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name uvrefl331 output.type <geotiff.argb jpeg.argb png.argb>	ompsrefl331-geotiff
ompsso2	OMPS SO ₂ concentration (Resolution 0.1 ⁰)	input.data <OMPS Total Column Total SO ₂ NRT HDF output (from OMPSnadir_SPA)> h2gout <Path to tif, png or jpeg output image file> config.type standard config.name ompsso2 output.type <geotiff.argb jpeg.argb png.argb>	ompsso2-geotiff

11.1.1 Generating H₂G Standard Products

Command line examples to generate standard H₂G products from the testscripts directory are provided below.

Example 1.1: MODIS TrueColor 0.01 degree tif (run from the testscripts directory)

```
$/wrapper/h2g/run \  
  config.type standard \  
  config.name tcolor0_01 \  
  input.data ../testdata/input/MYDcrefl.08085185938.hdf \  
  geo ../testdata/input/MYD03.08085185938.hdf \  
  h2gout ../testdata/output/TCOLOR0.01.tif \  
  output.type geotiff.rgb
```

Example 1.2: VIIRS SDR Top of Atmosphere True Color (run from the testscripts/ directory)

```
$/wrapper/h2g/run \  
  config.type standard \  
  config.name vtcolor \  
  red.data ../testdata/input/SVM05_npp_d20130323_t1851552_e1853194_b07270_c20130329144411503651_noaa_ops.h5 \  
  green.data ../testdata/input/SVM04_npp_d20130323_t1851552_e1853194_b07270_c20130329144448698975_noaa_ops.h5 \  
  blue.data ../testdata/input/SVM03_npp_d20130323_t1851552_e1853194_b07270_c20130329144447345002_noaa_ops.h5 \  
  geo ../testdata/input/GMTCO_npp_d20130323_t1851552_e1853194_b07270_c20130329144438416689_noaa_ops.h5 \  
  h2gout ../testdata/output/VTOA.png \  
  output.type png.rgb
```

NOTE: Example 1.2 shows how to run H₂G when the red, green and blue bands needed to create an RGB image are in different files

Example 2: MODIS EVI 0.01 degree jpeg (run from the testscripts directory)

```
$/wrapper/h2g/run \  
  config.type standard \  
  config.name evi \  
  input.data ../testdata/input/MYD013.08085185938.hdf \  
  geo ../testdata/input/MYD03.08085185938.hdf \  
  mask ../testdata/input/MYD014.08085185938.hdf \  
  h2gout ../testdata/output/EVI.jpg \  
  output.type jpeg.rgb
```

A successful execution usually takes some time, so if the execution seems to get stuck, do not become impatient. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct the problem and run again. The problem can also be identified using the `stdfile*` and `errfile*` log files. Log files are automatically generated within the directory used for execution.

NOTES:

1. Command line examples for generating H₂G standard outputs are provided in the run-h2g-standard-output file within the testscripts directory. All command lines are commented. Please uncomment the desired command line and then type `./run-h2g-standard-input` to execute the command. H₂G standard output products are available for comparison in the `testdata/output/h2g_standard_outputs` subdirectory.
2. The `tcolor0_01/tcolorfire0_01/ndvi/evi` standard products are optimized for ~1km true color images. For higher resolution true color images use the `tcolor0_0025`, `tcolor0_05`, `tcolorfire0_0025`, `tcolorfire0_05`, `ndvi0_0025`, `evi0_0025` products. Users are cautioned that processing of the higher resolution true-color products is resource-intensive and may fail if there is insufficient memory/processing power on your computer.
3. The IMAPP aerosol product may produce insufficient geolocation data at higher latitudes. H₂G would fail to produce correct aerosol image products in such cases.
4. Please see Appendix D for additional information on standard products, such as: (a) how the products were scaled into 8-bit images; (b) how the masks were used; and (c) how to retrieve actual parameter values from the GeoTIFF images.

11.1.2 Overriding Projection and Resolution of Standard Products

H₂G standard products are by default in geographic projection. In order to override this default projection and use any other projection (currently the only other projection is stereographic), you should use the projection and resolution parameter labels on the command line. Note that the geographic projection requires resolution in latitude/longitude degree units, while the stereographic projection requires resolution in meter units. You may also override only the resolution label to get a geographically projected image in a different resolution.

Example 3: MODIS TrueColor Stereographic 1000 meter tif (run from the testscripts directory)

```
$/./wrapper/h2g/run \  
  config.type standard \  
  config.name tcolor0_01 \  
  input.data ../testdata/input/MYDcrefl.08085185938.hdf \  
  fireloc ../testdata/input/MYD014.08085185938.txt \  
  geo ../testdata/input/MYD03.08085185938.hdf \  
  h2gout ../testdata/output/TCOLOR0.01_stereo.tif \  
  output.type geotiff.rgb \  
  projection stereographic \  
  resolution 1000
```

Example 4: MODIS EVI Stereographic 1000m png (run from the testscripts directory)

```
$/./wrapper/h2g/run \  
  config.type standard \  
  config.name evi \  
  input.data ../testdata/input/MYD013.08085185938.hdf \  
  geo ../testdata/input/MYD03.08085185938.hdf \  
  mask ../testdata/input/MYD014.08085185938.hdf \  
  h2gout ../testdata/output/EVI_stereo.png \  
  output.type png.rgb \  
  projection stereographic \  
  resolution 1000
```

Example 5: MODIS EVI Geographic 0.05 degree thumbnail png (run from the testscripts directory)

```
$ ./wrapper/h2g/run \  
  config.type standard \  
  config.name evi \  
  input.data ../testdata/input/MYD013.08085185938.hdf \  
  geo ../testdata/input/MYD03.08085185938.hdf \  
  mask ../testdata/input/MYD014.08085185938.hdf \  
  h2gout ../testdata/output/EVI_thumbnail.png \  
  output.type png.rgb \  
  resolution 0.05
```

NOTES:

1. The center of projection for the stereographic projection (for non-subset and non-mosaic images) is automatically selected by H₂G at approximately the

midpoint of the image.

2. When you override the default projection, be sure to also specify the resolution label along with the projection label. Care should be taken while specifying the resolution, as the resolution units of the geographic (degrees) and stereographic (meters) projections are different.

11.2 User-defined Products

H₂G is pre-configured to generate standard image products only. To generate user-defined outputs, the user must write a configuration file in the correct format. The user must then set the config.name label to the configuration file path and the config.type label to either 'singleband' or 'rgb'. Detailed instructions for writing configuration files and using them in command lines follow.

11.2.1 Configuration Files for User-defined Products

In order to generate non-standard products the user must create a configuration file describing the desired image product. The configuration file must set values for various configuration keywords. There are two types of configuration files: configuration files for single-band images, and configuration files for RGB images. Descriptions of these two types of configuration files follow.

NOTE: Configuration file examples are provided in the testscripts/more_examples and testscripts/other_examples directories. Command line examples for using the configuration files are provided in the run-xxx files within these directories. All command lines are commented. Please uncomment the desired command line and then type `./run-xxx` to execute the command. Some H₂G user-defined products are available for comparison in the testdata/output/more_examples and testdata/output/other_examples subdirectories.

11.2.1.1 Configuration Files for User-defined Single-band Images

Example 6: Single-band configuration file (MODIS LST with an NDVI mask)

The following configuration file 'lstndvimaskconfig.txt' is available in testscripts/more_examples.

```
DATASET      RR_LST_1KM
GEOLOCATION   MODISL1DB_MOD03_LAT_1KM MODISL1DB_MOD03_LON_1KM
GEOSOURCE   GEOFILE
MASK        RR_NDVI_250M CONTINUOUS 2 -1000 2500
MASKSOURCE  MASKFILE
SCALING     LINEAR 2300 3400
COLORMAP    ./colormap.txt
LEGEND      LST(degK) 5 1 230 63 257.5 128 285 191 312.5 255 340
PROJECTION  GEOGRAPHIC 0.01
SCANS_PER_LOOP  10
```

NOTE: There should be no empty lines (even after the last line) in the configuration file.

Example 7: Running H₂G using the single-band configuration file (run from the testscripts/more_examples directory)

```
$.../wrapper/h2g/run \  
    config.type singleband \  
    config.name ./lstndvimaskconfig.txt \  
    input.data ../testdata/input/LST.08085185938.hdf \  
    mask ../testdata/input/MYD013.08085185938.hdf \  
    geo ../testdata/input/MYD03.08085185938.hdf \  
    h2gout ../testdata/output/LST_NDVIMASK.jpg \  
    output.type jpeg.argb
```

A successful execution usually takes some time (approximately 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck, do not become impatient. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct the problem and run again. The problem can also be identified using the `stdfile*` and `errfile*` log files. Log files are automatically generated within the directory used for execution.

11.2.1.2 Keyword Descriptions for Single-band User-defined Image Configuration Files

DATASET *DatasetKey*

[Mandatory] The DATASET keyword must be followed by a *DatasetKey*. A *DatasetKey* is an identifier for the dataset for which you want to generate the image. The dataset must be present in the HDF file given to H₂G via the `input.data` command line label. Appendix A lists the DatasetKeys for the datasets and the MODIS/VIIRS products currently supported by H₂G.

GEOLOCATION *LatitudeDatasetKey LongitudeDatasetKey*

[Mandatory] The GEOLOCATION keyword identifies the geolocation dataset that will be used to project the image. Appendix B lists the LatitudeDatasetKeys and LongitudeDatasetKeys, and their HDF sources. For MODIS use the MOD03 geolocation keys if the L2 HDF files do not contain any geolocation information (e.g., HDF outputs from the NDVIEVI_SPA do not contain latitude/longitude datasets). Otherwise use the geolocation datasets that are already included within the `input.data` HDF product (e.g., HDF outputs from IMAPP_SPA have their own geolocation datasets). Keys for VIIRS geolocation products in different resolutions (GMTCO, GDNBO or GITCO) are also available for VIIRS inputs.

GEOSOURCE `<DATASETFILE | GEOFILE>`

[Mandatory] The GEOSOURCE keyword is used to tell H₂G if the geolocation

datasets are included within the primary input.data HDF dataset or must be extracted from a separate HDF file (like the MOD03 dataset). GEOSOURCE can take one of the two values: DATASETFILE or GEOFILE. If the GEOSOURCE value is DATASETFILE, H₂G understands that the geolocation datasets are inside the HDF file identified by the input.data label. In this case the geo label need not be used on the command line. If the GEOSOURCE value is GEOFILE, H₂G extracts the geolocation datasets from a separate file identified to it via the geo command line label.

MASK DatasetKey <DISCRETE|CONTINUOUS> #NoOfValues #Value1 #Value2

[Optional] The MASK keyword identifies the dataset used as a mask for the image product. It must be followed by a DatasetKey. (Appendix A lists the DatasetKeys for the datasets currently supported by H₂G). The next few values describe how the mask is to be used. The first value after the DatasetKey identifies whether the mask is DISCRETE or CONTINUOUS. A mask is DISCRETE when you want to mask areas where the mask dataset has certain discrete values. A mask is CONTINUOUS when you want to mask areas where the mask dataset values are within a particular continuous range. The next few values identify which values or ranges to mask. Examples to clarify the use of this keyword are provided below.

Example :

```
MASK RR_FIREMASK_1KM DISCRETE 2 3 4
```

This indicates that the RR_FIREMASK_1KM product will be used as a mask. The mask is discrete, meaning some discrete values need to be masked. The '2' tells H₂G how many values need to be masked; in this case two values of '3' and '4' are to be masked ('3' and '4' represent cloud and water respectively in the RR_FIREMASK_1KM dataset).

Example :

```
MASK RR_LST_1KM CONTINUOUS 4 2000 2700 3000 3500
```

This indicates that the RR_LST_1KM product will be used as a mask. The mask is continuous, meaning values within some ranges need to be masked (more than one range can be specified). The first '4' tells H₂G how many values follow after it. Here LST values in the dataset between two ranges, 2000-2700 and 3000-3500, will be masked. Note that the ranges should be specified in terms of the actual values inside the dataset and not the parameter values. Since the LST dataset appears with a scaling factor of 0.1 inside the HDF file, you should use 2000-2700 to mean 200K-270K.

MASKSOURCE <DATASETFILE | MASKFILE>

[Mandatory when the MASK keyword is used] The MASKSOURCE keyword is

used to tell H₂G if the mask datasets are included within the primary HDF dataset, or if they must be extracted from a separate HDF file. MASKSOURCE can take one of the two values, DATASETFILE or MASKFILE. If the MASKSOURCE value is DATASETFILE, H₂G understands that the mask dataset is available within the HDF file identified by the input.data label. In this case the mask label need not be used on the command line. If the MASKSOURCE value is MASKFILE, H₂G extracts the mask dataset from a separate file identified to it via the mask command line label.

FIRESOURCE FIREFILE

[Optional] The FIRESOURCE keyword is used to overlay fire pixels on browse images. Note that the FIRESOURCE keyword is ignored when the output.type is geotiff.u8cm. If you use the FIRESOURCE keyword in your configuration file, you must provide the firelocation text file (output from the MOD14_SPA) as input to H₂G via the fireloc command line label.

SCALING <LINEAR| SEGMENTED_LINEAR | NONLINEAR> *ScalingParameters*

[Mandatory] The SCALING keyword describes the scaling used to transform the dataset values in the input HDF file into 8-bit values between 0 and 255 in the output image. There are three types of SCALING that can be used. The number and type of *ScalingParameters* differs depending on the type of SCALING employed. SCALING types and examples are provided below.

LINEAR *minValue maxValue*

Example:

```
SCALING      LINEAR -1000 10000
```

The HDF values between min*Value* and max*Value* will be scaled linearly between 1 and 255. Any value < min*Value* will be rounded up to 1. Any value > max*Value* will be rounded down to 255. FillValues will be set as 0 (see Appendix A for FillValues for each dataset). In this example values between -1000 and 10000 will be scaled linearly between 1 and 255.

SEGMENTED_LINEAR *noOfTiePoints < hdfValue imageValue >**

Example:

```
SCALING      SEGMENTED_LINEAR 3 -1000 1 2500 50 7500 255
```

The SEGMENTED_LINEAR scaling allows the user to dedicate more color variations within a desired range. In this example, the scale has 3 tie-points and two linear segments. Values between -1000 and 2500 will be scaled linearly between 1 and 50, and values between 2500 and 7500 will be scaled linearly between 50 and 255.

NONLINEAR *c0 c1 c2 c3 d*

Example:

```
NONLINEAR -253 0 0 0 63.5
```

The scaling will be done using the following function:

geotiff value= $c0+c1*hdvalue+c2*hdvalue^2+c3*hdvalue^3+d*\log_{10}(hdvalue)$

Any value above 255 will be rounded down to 255 and any value below 1 will be rounded up to 1.

COLORMAP <Standard_ColorMap | PathToColorMapFile>

[Mandatory] The COLORMAP keyword identifies an 8-bit colormap for the image product. The colormap has 256 RGB triplets, one for each possible pixel value from 0 to 255. Users can use either a standard colormap by specifying its identifier, or specify a path to their own colormap file. Standard color map identifiers are as follows:

- a) NDVI
- b) LST
- c) FIRE
- d) SST
- e) CHLOR
- f) AOD
- g) IRPHASE
- h) CTP
- i) CTT
- j) TPW
- k) CLOUDMASK
- l) IMGEDR
- m) VCLOUDPHASE
- n) VCLOUDMASK
- o) VSNOWBIN
- p) VSUSPMAT
- q) VSURFALBEDO
- r) BURNSCAR
- s) GRAYSCALE
- t) UVAEROSOL
- u) TO3
- v) UVREFL

w) SO2

A user-defined colormap file must conform to a particular format. An example is contained in Figure 1.

#idx	R	G	B
0	0	0	0
1	144	0	111
2	141	0	114
...
...
...
255	100	0	0

Figure 1. User-defined Colormap File Example

NOTE: See colormap file format in testscripts/other_examples/colormap.txt.

LEGEND *LegendString NoOfTicks <TickPosition TickString>**

[Optional] The LEGEND keyword can be used to burn a legend into a jpeg or png browse image. Note that the LEGEND keyword is used only when the output.type is either jpeg.rgb or png.rgb; it is ignored otherwise. The legend shows 256 colors corresponding to the colormap (standard or user-defined).

Example:

```
LEGEND      Cloudbottom_Pressure(hPa) 5 1 10 64 274 128 550 192 827
255 1100
```

The legend string is Cloudbottom_Pressure(hPa). (Note that no blanks are allowed within the legend string.) The legend shows 255 colors and has 5 ticks specified by the first 5 after the legend string. The string 10 is burned in at position 1, 274 at position 64, and so on. The legend would be burned in at the left bottom corner of the image and would appear as shown in Figure 2.

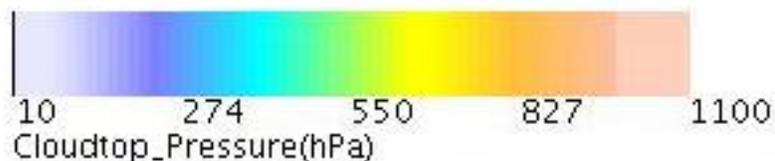


Figure 2. Legend Example

PROJECTION GEOGRAPHIC *Resolution*

[Mandatory] The PROJECTION keyword specifies the projection and resolution of the output image. Resolution should be specified in degrees. Currently only the 'GEOGRAPHIC' projection is supported within a configuration file. However,

you can always override the default projection and resolution on the command line (see paragraph 11.2.2).

NOTE: A geographic projection may not be suitable for very high latitudes. Use stereographic projections instead.

SCANS_PER_LOOP *NoOfScansProcessedAtATime*

[Mandatory] This is a processing parameter that can be adjusted based on the memory requirements of the output product. For example if memory is low, you can set SCANS_PER_LOOP to 10, meaning 10 scans will be processed at a time, thus requiring less memory. Setting SCANS_PER_LOOP to low values is useful when generating high-resolution images that require more memory. However, setting SCANS_PER_LOOP to low values also slows execution.

NO_INTERPOLATION #NoOfValues <Values>*

[Optional] This keyword may be useful when certain values should not be used for interpolation during projection. This keyword's utility is illustrated by the following example. The values '7', '8' and '9' within the RR_FIREMASK_1KM dataset in the MOD14 HDF product denote fire pixels. It is desirable that during projection of the image product, these fire pixel values not be used for interpolation; otherwise, there may be spurious fire pixels in the image product resulting from interpolation.

Example:

NO_INTERPOLATION 3 7 8 9

3 values, namely '7', '8' and '9', should not be used for interpolation during creation of the projected image.

11.2.1.3 Configuration Files for User-defined RGB Images

Example 8: RGB configuration file (False Color Image from the MODIS Corrected Reflectance HDF product with fire pixel overlays):

The following configuration file 'fcolorconfig.txt' is available in testscripts/other_examples.

```

REDDATASET      CREFL250M_BAND2
REDSCALE        6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
GREENDATASET    CREFL250M_BAND1
GREENSCALE      6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
BLUEDATASET     CREFL250M_BAND4
BLUESCALE       6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
GEOLOCATION       MODISL1DB_MOD03_LAT_1KM MODISL1DB_MOD03_LON_1KM
GEOSOURCE       GEOFILE
FIRESOURCE      FIREFILE
PROJECTION       GEOGRAPHIC 0.01
SCANS_PER_LOOP  10

```

NOTE: There should be no empty lines (even after the last line) in the configuration file.

Example 9: Running H₂G using the RGB configuration file (run from the testscripts/other_examples directory)

```

$../wrapper/h2g/run \
  config.type rgb \
  config.name ./fcolorconfig.txt \
  input.data ../testdata/input/MYDcrefl.08085185938.hdf \
  fireloc ../testdata/input/MYD014.08085185938.txt \
  geo ../testdata/input/MYD03.08085185938.hdf \
  h2gout ../testdata/output/FCOLOR.jpg \
  output.type jpeg.rgb

```

A successful execution usually takes some time (approximately 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck, do not become impatient. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct the problem and run again. The problem can also be identified using the `stdfile*` and `errfile*` log files. Log files are automatically generated within the directory used for execution.

11.2.1.4 Keyword Descriptions for RGB User-defined image Configuration Files

REDDATASET *DatasetKey*

[Mandatory] The REDDATASET keyword specifies the dataset used as the Red band in the output RGB image. A *DatasetKey* is an identifier for the dataset you wish to use as the Red band. The dataset must be present in the HDF file given to H₂G via the `input.data` or the `red.data` command line label. Appendix A lists the DatasetKeys for the various datasets.

REDSCALE *noOfTiePoints [hdfValue imageValue]**

[Mandatory] The REDSCALE keyword describes the scaling used to transform the red dataset values in the input HDF file into 8-bit values between 0 and 255. A segmented linear scaling is used for standard RGB images.

Example:

```
REDSCALE    6 0 0 1294 110 2588 160 5176 210 8196 240 11000 255
```

In the above example, the scale has 6 tie-points (and five linear segments) identified by the first 6 after the keyword. Values between 0 and 1294 will be scaled linearly between 0 and 110; values between 1294 and 2588 will be scaled linearly between 110 and 160; values between 2588 and 5176 will be scaled linearly between 160 and 210; values between 5176 and 8196 will be scaled linearly between 210 and 240; and values between 8196 and 11000 will be scaled linearly between 240 and 255.

GREENDATASET *DatasetKey*

[Mandatory] The GREENDATASET keyword specifies the dataset used as the Green band in the output RGB image. A *DatasetKey* is an identifier for the dataset you wish to use as the Green band. The dataset must be present in the HDF file given to H₂G via the input.data or the green.data command line label. Appendix A lists the DatasetKeys for the various datasets.

GREENSCALE *noOfSegments [hdfValue imageValue]**

[Mandatory] The GREENSCALE keyword describes the scaling used to transform the green dataset values in the input HDF file into 8-bit values between 0 and 255. A segmented linear scaling is used for standard RGB images.

BLUEDATASET *DatasetKey*

[Mandatory] The BLUEDATASET keyword specifies the dataset used as the Blue band in the output RGB image. A *DatasetKey* is an identifier for the dataset you want to use as the Blue band. The dataset must be present in the HDF file given to H₂G via the input.data or the blue.data command line label. Appendix A lists the DatasetKeys for the various datasets.

BLUESCALE *noOfSegments [hdfValue imageValue]**

[Mandatory] The BLUESCALE keyword describes the scaling used to transform the blue dataset values in the input HDF file into 8-bit values between 0 and 255. A segmented linear scaling is used for standard RGB images.

GEOLOCATION *LatitudeDatasetKey LongitudeDatasetKey*

[Mandatory] See configuration file entries for single-band images in paragraph 11.2.1.2.

GEOSOURCE *<DATASETFILE | GEOFILE>*

[Mandatory] See configuration file entries for single-band images in paragraph

11.2.1.2.

FIRESOURCE FIREFILE

[Optional] See configuration file entries for single-band images in paragraph 11.2.1.2.

PROJECTION GEOGRAPHIC *Resolution*

[Mandatory] See configuration file entries for single-band images in paragraph 11.2.1.2.

SCANS_PER_LOOP *NoOfScansProcessedAtATime*

[Mandatory] See configuration file entries for single-band images in paragraph 11.2.1.2.

NOTE: The REDDATASET, BLUEDATASET and the GREENDATASET may exist in the same HDF input file or they may exist in different HDF files. If they exist in the same HDF file, the file must be provided to H₂G via the input.data command line label as described in the above example. If they exist in separate files, this must be specified in the command line using the red.data, green.data and blue.data input labels (see Table 1 and Example 1.2).

11.2.2 Overriding Projection and Resolution for User-defined Products

In order to override the projection specified in the user-defined configuration file you should use the projection and resolution parameter labels on the command line. Note that the geographic projection requires resolution in latitude/longitude degree units, while the stereographic projection requires resolution in meter units. You may also override only the resolution label to modify just the resolution of the output image.

Example 10: Overriding projection and resolution defined in the configuration file (run from the testscripts/more_examples directory)

```
$../wrapper/h2g/run \  
  config.type singleband \  
  config.name ./lstndvimaskconfig.txt \  
  input.data ../testdata/input/LST.08085185938.hdf \  
  mask ../testdata/input/MYD013.08085185938.hdf \  
  geo ../testdata/input/MYD03.08085185938.hdf \  
  h2gout ../testdata/output/LST_NDVIMASK_STEREO.jpg \  
  output.type jpeg.rgb \  
  projection stereographic \  
  resolution 1000
```

11.3 Subsetting and Mosaicing

Subsetting in H₂G is the extraction of a region of interest from an input swath for purposes of creating an output image product. Mosaicing in H₂G is the stitching of

multiple input swaths for purposes of creating a concatenated output image product. In H₂G, subsetting and mosaicing use a similar paradigm. To perform these operations users must define a region of interest. This is achieved by specifying an additional set of parameter labels on the command line. The labels 'centerlat' and 'centerlon', used for both geographic and stereographic projections, specify the center of the output image in latitude and longitude degrees respectively. (For stereographic projections these labels also define the center of projection). The labels 'height_lat' and 'width_lon', used only for geographic projections, specify the width and height of the image in latitude and longitude degrees. However for stereographic projections, the 'height_km' and 'width_km' labels are used (instead of the 'height_lat' and 'width_lon' labels) to specify the vertical and horizontal extents of the output image in kilometers.

Mosaicing is an extension of subsetting. Users need to employ the additional input.datan, geon, maskn, firelocn, red.datan, blue.datan, green.datan (n=1,2,...10) labels as necessary to specify multiple swath datasets to H₂G.

NOTE: More command line examples for subsetting and mosaicing can be found in the run-subset-egs and run-mosaic-egs scripts in testscripts/more_examples. All command lines are commented. Uncomment the desired command line and then type `./run-subset-egs` or `./run-mosaic-egs` to execute the command. Some H₂G subsetting/mosaiced output products are available for comparison in the testdata/output/more_examples subdirectory.

NOTE: Users can provide up to 10 swaths for mosaicing using the input.datan, geon, maskn, firelocn, red.datan, blue.datan, green.datan labels.

Example 11: Subsetting MODIS TrueColor Geographic 0.01 degree tif (run from the testscripts/more_examples directory)

```
$ ../../wrapper/h2g/run \  
  config.type standard config.name tcolor0_01 \  
  input.data ../../testdata/input/MYDcrefl.08085185938.hdf \  
  geo ../../testdata/input/MYD03.08085185938.hdf \  
  h2gout ../../testdata/output/TCOLOR_SUBSET.tif \  
  output.type geotiff.rgb \  
  centerlat 45.0 centerlon -95.0 height_lat 20.0 width_lon 20.0
```

Example 12: Subsetting user-defined MODIS Cloudtop Pressure Stereographic 1000m png (run from the testscripts/more_examples directory)

```
$ ../../wrapper/h2g/run \  
  config.type singleband config.name ./ctpirphasemaskconfig.txt \  
  input.data ../../testdata/input/CLOUDTOP.08085185938.hdf \  
  geo ../../testdata/input/MYD03.08085185938.hdf \  
  h2gout ../../testdata/output/CTPTOP.08085185938.png
```

```
h2gout ../testdata/output/CTP_IRPHASEMASK_STEREO_SUBSET.png \  
output.type png.argb \  
projection stereographic resolution 5000 \  
centerlat 45.0 centerlon -95.0 height_km 2000 width_km 2000
```

Example 13: Mosaicing VIIRS Top-of-Atmosphere TrueColor Stereographic 750m tif (run from the testscripts/more_examples directory)

```
$.../wrapper/h2g/run config.type standard config.name vtcOLOR \  
red.data ../testdata/input/SVM05_npp_d20120925_t1804560_e1817535.h5 \  
green.data ../testdata/input/SVM04_npp_d20120925_t1804560_e1817535.h5 \  
blue.data ../testdata/input/SVM03_npp_d20120925_t1804560_e1817535.h5 \  
geo ../testdata/input/GMTCO_npp_d20120925_t1804560_e1817535.h5 \  
red.data2  
../testdata/input/SVM05_npp_d20130323_t1851552_e1853194_b07270_c201303291444  
11503651_noaa_ops.h5 \  
green.data2  
../testdata/input/SVM04_npp_d20130323_t1851552_e1853194_b07270_c201303291444  
48698975_noaa_ops.h5 \  
blue.data2  
../testdata/input/SVM03_npp_d20130323_t1851552_e1853194_b07270_c201303291444  
47345002_noaa_ops.h5 \  
geo2  
../testdata/input/GMTCO_npp_d20130323_t1851552_e1853194_b07270_c201303291444  
38416689_noaa_ops.h5 \  
h2gout ../testdata/output/VTOA_STEREO_MOSAIC.tif \  
output.type geotiff.argb \  
projection stereographic resolution 750 \  
centerlat 38.99 centerlon -76.85 \  
height_km 6000 width_km 6000
```

Example 14: Mosaicing MODIS TrueColor Geographic 0.01 degree png (run from the testscripts/more_examples directory)

```
$.../wrapper/h2g/run config.type standard config.name tcolorfire0_01 \  
input.data ../testdata/input/MYDcrefl.12233171444.hdf \  
geo ../testdata/input/MYD03.12233171444.hdf \  
input.data2 ../testdata/input/MYDcrefl.12233185223.hdf \  
geo2 ../testdata/input/MYD03.12233185223.hdf \  
fireloc2 ../testdata/input/MYD14.12233185223.txt \  
h2gout ../testdata/output/CREFL_GEO_MOSAIC.png \  
output.type png.argb \  
centerlat 38.99 centerlon -76.85 \  
height_lat 45 width_lon 45
```

Appendix A Dataset Identifiers

Dataset Identifier	Corresponding Scientific Data Set (SDS)	Fill Value	SPA Producing the HDF Product That Contains the SDS
RR_NDVI_250M	NDVI	-999	NDVIEVI_SPA
RR_EVI_250M	EVI	-999	NDVIEVI_SPA
RR_FIREMASK_1KM	fire mask	0	MOD14_SPA
RR_LST_1KM	LST	0	MODLST_SPA
RR_T31_1KM	T31	0	MODLST_SPA
RR_T32_1KM	T32	0	MODLST_SPA
MSL12_SST_1KM	sst	-32767	L2GEN_SPA (MODIS SST)
MSL12_QUALSST_1KM	qual_sst	0	L2GEN_SPA (MODIS SST)
MSL12_CHLORA_1KM	chlor_a Note: The original floating point HDF values are multiplied by 10 ⁶ to create integers before H ₂ G uses them for further processing.	-32767 to -1	L2GEN_SPA (MODIS Chlor)
MSL12_L2FLAGS_1KM	l2flags Note: The original values in HDF are bit fields. They are converted into integer values by H ₂ G for convenience. The values are: 0-Chlor_Ok; 1-Chlor_Warning ; 2-Chlor_Fail. MSL12_L2FLAGS_1KM can only be used as a mask.	(see Notes below)	L2GEN_SPA (MODIS Chlor)
IMAPP_AOD_10KM	Optical_Depth_Land_And_Ocean	-9999	IMAPP_SPA (mod04)
IMAPP_ODR_10KM	Optical_Depth_Ratio_Small_Land_And_Ocean	-9999	IMAPP_SPA (mod04)
IMAPP_CLOUDPHASE_5KM	Cloud_Phase_Infrared	127	IMAPP_SPA (mod06)

IMAPP_CTP_5KM	Cloud_Top_Pressure	-32768	IMAPP_SPA (mod06)
IMAPP_CTT_5KM	Cloud_Top_Temperature	-32768	IMAPP_SPA (mod06)
IMAPP_CLFR_5KM	Cloud_Fraction	127	IMAPP_SPA (mod06)
IMAPP_CEMS_5KM	Cloud_Effective_Emissivity	127	IMAPP_SPA (mod06)
IMAPP_STYPE_5KM	Surface_Type	-32768	IMAPP_SPA (mod06)
IMAPP_STEMP_5KM	Surface_Temperature	-32768	IMAPP_SPA (mod06)
IMAPP_SPRES_5KM	Surface_Pressure	-32768	IMAPP_SPA (mod06)
IMAPP_SELEV_5KM	Surface_Elevation	-32768	IMAPP_SPA (mod07)
IMAPP_TRHGT_5KM	Tropopause_Height	-32768	IMAPP_SPA (mod06)
IMAPP_TPW_5KM	Water_Vapor	-9999	IMAPP_SPA (mod07)
IMAPP_OZONE_5KM	Total_Ozone	-32768	IMAPP_SPA (mod07)
IMAPP_TOTALS_5KM	Total_Totals	-32768	IMAPP_SPA (mod07)
IMAPP_LIFTEDIDX_5KM	Lifted_Index	-32768	IMAPP_SPA (mod07)
IMAPP_KIDX_5KM	K_Index	-32768	IMAPP_SPA (mod07)
IMAPP_TPWD_5KM	Water_Vapor_Direct	-9999	IMAPP_SPA (mod07)
IMAPP_CLOUDMASK_1KM	Cloud Mask Note: The original values in HDF are bit fields. They are converted into integer values by H ₂ G for convenience. The values are: 0-No Value; 1-Cloudy; 2-Uncertain; 3-Probably Clear; 4-Clear.	0 (see Note below)	IMAPP_SPA (mod35)
CREFL250M_BAND1	CorrRefl_01	32767	CREFL_SPA (default resolution)
CREFL250M_BAND2	CorrRefl_02	32767	CREFL_SPA (default resolution)
CREFL500M_BAND3	CorrRefl_03	32767	CREFL_SPA (default resolution)

CREFL500M_BAND4	CorrRefl_04	32767	CREFL_SPA (default resolution)
CREFL500M_BAND5	CorrRefl_05	32767	CREFL_SPA (default resolution)
CREFL500M_BAND6	CorrRefl_06	32767	CREFL_SPA (default resolution)
CREFL500M_BAND7	CorrRefl_07	32767	CREFL_SPA (default resolution)
MOD09_250MBAND1	Surface Reflectance Band 1	-28672	MOD09_SPA
MOD09_250MBAND2	Surface Reflectance Band 2	-28672	MOD09_SPA
MOD09_500MBAND3	Surface Reflectance Band 3	-28672	MOD09_SPA
MOD09_500MBAND4	Surface Reflectance Band 4	-28672	MOD09_SPA
MOD09_500MBAND5	Surface Reflectance Band 5	-28672	MOD09_SPA
MOD09_500MBAND6	Surface Reflectance Band 6	-28672	MOD09_SPA
MOD09_500MBAND7	Surface Reflectance Band 7	-28672	MOD09_SPA
CREFLTILE_BAND1	CorrRefl_01 (from MODIS Daily Composite Corrected Reflectance product)	32767	BURNSCAR_SPA
CREFLTILE_BAND2	CorrRefl_02 (from MODIS Daily Composite Corrected Reflectance product)	32767	BURNSCAR_SPA
CREFLTILE_BAND3	CorrRefl_03 (from MODIS Daily Composite Corrected Reflectance product)	32767	BURNSCAR_SPA
CREFLTILE_BAND4	CorrRefl_04 (from MODIS Daily Composite Corrected Reflectance product)	32767	BURNSCAR_SPA
CREFLTILE_BAND5	CorrRefl_05 (from MODIS Daily Composite Corrected Reflectance product)	32767	BURNSCAR_SPA
CREFLTILE_BAND6	CorrRefl_06 (from MODIS Daily Composite Corrected Reflectance product)	32767	BURNSCAR_SPA
CREFLTILE_BAND7	CorrRefl_07 (from MODIS Daily Composite Corrected Reflectance product)	32767	BURNSCAR_SPA

	Reflectance product)		
MOD14TILE	FireMask	0	BURNSCAR_SPA
NPP_M12_H5	/All_Data/VIIRS-M12-SDR_All/BrightnessTemperature (from VIIRS SVM12 SDR)	65528 to 65535	C-SDR_SPA
NPP_M5_H5	/All_Data/VIIRS-M5-SDR_All/Reflectance (from VIIRS SVM05 SDR)	65528 to 65535	C-SDR_SPA
NPP_M4_H5	/All_Data/VIIRS-M4-SDR_All/Reflectance (from VIIRS SVM04 SDR)	65528 to 65535	C-SDR_SPA
NPP_M3_H5	/All_Data/VIIRS-M3-SDR_All/Reflectance (from VIIRS SVM03 SDR)	65528 to 65535	C-SDR_SPA
NPP_DNB_H5	/All_Data/VIIRS-DNB-SDR_All/Radiance (from VIIRS SVDNB SDR) Note: The original floating point dataset values are multiplied by 10^5 to create integers before H ₂ G uses them for further processing.	-1000 to -999	C-SDR_SPA
NPP_DNBNIGHT_H5	/All_Data/VIIRS-DNB-SDR_All/Radiance (from VIIRS SVDNB SDR) Note: The original floating point dataset values are multiplied by 10^9 to create integers before H ₂ G uses them for further processing)	-1000 to -999	C-SDR_SPA
NPP_SREFL_M5_H5D	/All_Data/VIIRS-Surf-Ref-IP_All/m5 (from VIIRS IVISR) Note: The original floating point dataset values are multiplied by 10^2 to create integers before H ₂ G uses them for further processing	-1000 to -999	SurfReflect_SPA
NPP_SREFL_M4_H5D	/All_Data/VIIRS-Surf-Ref-IP_All/m4 (from VIIRS IVISR) Note: The original floating point	-1000 to -999	SurfReflect_SPA

	dataset values are multiplied by 10^2 to create integers before H ₂ G uses them for further processing		
NPP_SREFL_M3_H5D	/All_Data/VIIRS-Surf-Refl-IP_All/m3 (from VIIRS IVISR) Note: The original floating point dataset values are multiplied by 10^2 to create integers before H ₂ G uses them for further processing	-1000 to -999	SurfReflect_SPA
CVIIRS_M5	CorrRefl_01 (M5 Corrected Reflectance from CVIIRS Moderate resolution product)	14000-32767	CVIIRS_SPA
CVIIRS_M4	CorrRefl_04 (M4 Corrected Reflectance from CVIIRS Moderate resolution product)	14000-32767	CVIIRS_SPA
CVIIRS_M3	CorrRefl_03 (M3 Corrected Reflectance from CVIIRS Moderate resolution product)	14000-32767	CVIIRS_SPA
CVIIRS_I1	CorrRefl_08 (I1 Corrected Reflectance from CVIIRS Imagery resolution product)	14000-32767	CVIIRS_SPA
IMAGERY_M1	/All_Data/VIIRS-M1ST-EDR_All/BrightnessTemperature OrReflectance (from VIIRS M1ST Imagery EDR)	65528 to 65535	GTMImagery_SPA
IMAGERY_M4	/All_Data/VIIRS-M2ND-EDR_All/BrightnessTemperature OrReflectance (from VIIRS M2ND Imagery EDR)	65528 to 65525	GTMImagery_SPA
IMAGERY_M9	/All_Data/VIIRS-M3RD-EDR_All/BrightnessTemperature OrReflectance (from VIIRS M3RD Imagery EDR)	65528 to 65535	GTMImagery_SPA
IMAGERY_I1	/All_Data/VIIRS-I1-IMG-EDR_All/Reflectance (from VIIRS VI1BO I1 Imagery EDR)	65528 to 65535	GTMImagery_SPA
IMAGERY_I2	/All_Data/VIIRS-I2-IMG-EDR_All/Reflectance (from VIIRS VI2BO I2 Imagery EDR)	65528 to 65535	GTMImagery_SPA

IMAGERY_I3	/All_Data/VIIRS-I3-IMG-EDR_All/Reflectance (from VIIRS VI3BO I3 Imagery EDR)	65528 to 65535	GTMIimagery_SPA
IMAGERY_NCC_ALBEDO	/All_Data/VIIRS-NCC-EDR_All/Albedo (from VIIRS VNCCO EDR)	65528 to 65535	GTMIimagery_SPA
VIIRSAF_FIREMASK	fire mask (from VIIRS-AF product)	0	VIIRSAF_SPA
NPP_FIREMASK_H5	All_Data/VIIRS-AF-EDR_All/fireMask (from VIIRS AVAFO product)	0	ACTIVEFIRES_SPA
NPP_VCM_CLOUDMASK_H5	/All_Data/VIIRS-CM-IP_All/QF1_VIIRSCMIP (from VIIRS IICMO product) Note: The original values in HDF are bit fields. They are converted into integer values by H ₂ G for convenience. The values are: 0-No Value; 1-Clear; 2-Probably Clear; 3-Uncertain; 4-Cloudy	0	CLOUDMASK_SPA
NPP_VCM_CLOUDPHASE_H5	/All_Data/VIIRS-CM-IP_All/QF6_VIIRSCMIP (from VIIRS IICMO product) Note: The original values in HDF are bit fields. They are converted into integer values by H ₂ G for convenience. The values are: 0-No Value; 1-Clear; 2-Partly Cloudy; 3-Water; 4-Mixed; 5-Ice; 6-Cirrus, 7 Cloud Overlap	0	CLOUDMASK_SPA
NPP_VAOT_550_H5	/All_Data/VIIRS-Aeros-Opt-Thick-IP_All/faot550 (from VIIRS IVAOT product) Note: The original floating point dataset values are multiplied by 10 ³ to create integers before H ₂ G uses them for further processing	-1000 to -999	AEROSOL_SPA
NPP_AERPRTCLSZ_H5	/All_Data/VIIRS-Aeros-Opt-Thick-IP_All/angexp (from VIIRS IVAOT product)	-1000 to -999	AEROSOL_SPA

	Note: The original floating point dataset values are multiplied by 10^3 to create integers before H ₂ G uses them for further processing		
NPP_SUSMATTYP_H5	/All_Data/VIIRS-SusMat-EDR_All/SuspendedMatterType (from VIIRS VSUMO product)	249 to 255	AEROSOL_SPA
NPP_VCOP_CTT_H5D	/All_Data/VIIRS-INWCTT-IP_All/ctt (from VIIRS IVIWT product) Note: The original floating point dataset values are multiplied by 10^2 to create integers before H ₂ G uses them for further processing	-1000 to -999	COP_SPA
NPP_VCOP_COT_H5D	/All_Data/VIIRS-Cd-Opt-Prop-IP_All/cot (from VIIRS IVCOP product) Note: The original floating point dataset values are multiplied by 10^2 to create integers before H ₂ G uses them for further processing	-1000 to -999	COP_SPA
NPP_VCOP_EPS_H5D	/All_Data/VIIRS-Cd-Opt-Prop-IP_All/eps (from VIIRS IVCOP product) Note: The original floating point dataset values are multiplied by 10^2 to create integers before H ₂ G uses them for further processing	-1000 to -999	COP_SPA
NPP_SNOWCOVER_H5	/All_Data/VIIRS-SCD-BINARY-SNOW-MAP-EDR_All/SnowCoverBinaryMap (from VIIRS VSCMO product)	249 to 255	SNOWCOV_SPA
NPP_SNOWCOVER_QUAL	/All_Data/VIIRS-SCD-BINARY-SNOW-MAP-EDR_All/QF1_VIIRSSCDBINARYSNOWMAPEDR Note: The original values in HDF are bit fields. They are converted into integer values by	0	SNOWCOV_SPA

	H ₂ G for convenience. The values are: 0-high quality; 1-medium quality; 2-bad quality; 3-no retrieval		
NPP_NDVI_H5	/All_Data/VIIRS-VI_EDR_All/TOA_NDVI (from VIIRS VIVIO product)	65528 to 65535	VEGINDEX_SPA
NPP_EVI_H5	/All_Data/VIIRS-VI_EDR_All/TOC_EVI (from VIIRS VIVIO product)	65528 to 65535	VEGINDEX_SPA
NPP_LST_H5	/All_Data/VIIRS-LST-EDR_All/LandSurfaceTemperature (from VIIRS VLSTO product)	65528 to 65535	LST_SPA
L2GEN_VSST	sst (from VIIRS L2GEN Sea Surface Temperature product)	-32767	L2GEN_SPA
L2GEN_VCHLORA	chlor_a (from VIIRS L2GEN Ocean Color product) Note: The original floating point HDF values are multiplied by 10 ⁶ to create integers before H ₂ G uses them for further processing.	-32767 to -1	L2GEN_SPA
L2GEN_VQUALSST	qual_sst (from VIIRS L2GEN SST product)	0	L2GEN_SPA
L2GEN_VL2FLAGS	l2_flags (from VIIRS L2GEN Ocean Color product) Note: The original values in HDF are bit fields. They are converted into integer values by H ₂ G for convenience. The values are: 0-Chlor_Ok; 1-Chlor_Warning ; 2-Chlor_Fail. MSL12_L2FLAGS_1KM can only be used as a mask	-9999	L2GEN_SPA
VVI_NDVI	NDVI	-999	VIIRS-VI_SPA
VVI_EVI	EVI	-999	VIIRS-VI_SPA
VIIRS-LST	LST	0	VIIRS-LST_SPA
OMPS_UVAEROSOL	/SCIENCE_DATA/UVAerosolIndex (from OMPS Total Column Total Ozone product)	-9999 to -3000	OMPSnadir_SPA

	Note: The original floating point HDF values are multiplied by 10^3 to create integers before H ₂ G uses them for further processing		
OMPS_TO3	/SCIENCE_DATA/ColumnAmountO3 (from OMPS Total Column Total Ozone product)	-9999 to 50	OMPSnadir_SPA
OMPS_REFL33	/SCIENCE_DATA/Reflectivity331 (from OMPS Total Column Total Ozone product) Note: The original floating point HDF values are multiplied by 10^3 to create integers before H ₂ G uses them for further processing	-9999 to -1	OMPSnadir_SPA
OMPS_SO2TRM	/HDFEOS/SWATHS/OMPS Column Amount SO2/Data Fields/ColumnAmountSO2_TRM (from OMPS Total Column Total SO ₂ NRT product) Note: The original floating point HDF values are multiplied by 10^3 to create integers before H ₂ G uses them for further processing	-9999 to -10	OMPSnadir_SPA

Notes on special datasets: H₂G usually reads in the unscaled SDS values from input HDF files. However, there are some datasets for which H₂G uses special conversions and/or scalings before any processing. These are noted in the above table. Users should use the appropriate SCALING parameters in configuration files when using them as primary datasets, or use the correct values when using them as masks.

Appendix B Geolocation Identifiers

Latitude Dataset Identifier	Longitude Dataset Identifier	SPA Producing the HDF Product That Contains the Scientific Data Set (SDS)
MODISL1DB_MOD03_LAT_1KM	MODISL1DB_MOD03_LON_1KM	MODISL1DB (mod03)
IMAPP_LAT_5KM	IMAPP_LON_5KM	IMAPP_SPA (mod06,mod07)
IMAPP_LAT_10KM	IMAPP_LON_10KM	IMAPP_SPA (mod04)
MOD09_LAT_1KM	MOD09LON1KM	MOD09_SPA
TILE_LAT_1KM	TILE_LON_1KM	BURNSCAR_SPA (Geolocation Tile product)
NPP_LAT_750_H5	NPP_LON_750_H5	VIIRS-SDR_SPA (Latitude and Longitude from the GMTCO Geolocation product)
NPP_LAT_375_H5	NPP_LON_375_H5	VIIRS-SDR_SPA (Latitude and Longitude from the GITCO 375m Geolocation product)
NPP_LAT_DNB_H5	NPP_LON_DNB_H5	VIIRS-SDR_SPA (Latitude and Longitude from the GDNBO DNB resolution Geolocation product)
IMAGERY_MLAT	IMAGERY_MLON	GTMI imagery_SPA (Latitude and Longitude from the GMGTO Geolocation product)
IMAGERY_ILAT	IMAGERY_ILON	GTMI imagery_SPA (Latitude and Longitude from the GIGTO Geolocation product)
IMAGERY_NCCLAT	IMAGERY_NCCLON	GTMI imagery_SPA (Latitude and Longitude from the GNCCO Geolocation product)
L2GEN_VLAT	L2GEN_VLON	L2GEN_SPA (Latitude and

		Longitude from the L2GEN VIIRS SST and Ocean Color products)
OMPS_LAT	OMPS_LON	OMPSnadir_SPA (Latitude and Longitude from the OMPS Total Column Total Ozone product)
OMPS_SO2LAT	OMPS_SO2LON	(Latitude and Longitude from the OMPS Total Column Total SO ₂ product)

Appendix C

Modifying Maximum Java Heap Size

To increase/decrease maximum Java heap size, cd into algorithm/bin and open the file h2g.sh. Edit the line '-Xmx2g' to the required value. For example, to decrease maximum Java heap size to 1G, edit it to '-Xmx1g'. To increase it to 4G, edit it to '-Xmx4g'.

CAUTION: Decreasing Java heap size may cause some high-resolution image generations to fail.

Appendix D

H₂G Standard Product Descriptions

This appendix describes the scaling used to convert Scientific Dataset (SDS) values in the SPA HDF products into 8-bit values in the standard GeoTIFF products. Pseudocodes for converting the GeoTIFF values back into actual parameter values are also provided. Please note that the parameter values obtained by inverse scaling GeoTIFF values will not be exactly equal to the parameter values obtained from actual SDS values (from the HDF products), but they should be close. Use of 8-bit integers in our GeoTIFF products may cause loss of precision. Further, values below and above the SDS data range being scaled into 1-255 are set to 1 and 255 respectively in the GeoTIFF output. Interpolation used during re-projection of swath data may also be a source of difference.

1. MODIS Aerosol Optical Depth

HDF SDS name: Optical_Depth_Land_And_Ocean (generated by IMAPP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0-5000 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-9999) in SDS are set to 0 in geotiff.

```
if hdf_value = -9999
  geotiff_value=0
else
  geotiff_value=1+round((254/5000)*hdf_value) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: 0-5):

```
if geotiff_value>0
  hdf_value=(geotiff_value-1)*5000/254 //scale 1-255 to 0-5000
  parameter_value=hdf_value*0.001 //apply scaling/offset factors
else //geotiff_value=0
  hdf_value=-9999
  parameter_value=NO_RETRIEVAL
end
```

2. MODIS Cloud Top Pressure

HDF SDS: Cloud_Top_Pressure (generated by IMAPP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 10-11000 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-32768) are set to 0.

```
if hdf_value = -32768
  geotiff_value=0
else
  geotiff_value=1+round((254/10990)*(hdf_value-10)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: hPa; range: 1-1100 hPa):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*10990/254]+10 //scale 1-255 to 10-11000
  parameter_value=hdf_value*0.1 //apply scaling and offset
else //geotiff_value=0
  hdf_value=-32768
  parameter_value=NO_RETRIEVAL
end
```

3. MODIS Cloud Phase

HDF SDS: Cloud_Phase_Infrared (generated by IMAPP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0 to 6 are scaled to 1-7 in GeoTIFF output. Fill Values (127) is set to 0.

```
if hdf_value=127
  geotiff_value=0
else
  geotiff_value=1+hdf_value
end
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter flags (units: dimensionless):

```
if(geotiff_value>0)
  hdf_value=geotiff_value-1
  parameter_flag=hdf_value (clear = 0; water = 1 or 5; ice = 2 or 4; mixed = 3; uncertain = 6)
else
  hdf_value=127
  parameter_flag=NO_RETRIEVAL
end
```

4. MODIS Total Precipitable Water

HDF SDS: Water_Vapor (generated by IMAPP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0-20000 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-9999) are set to 0.

```
if hdf_value = -9999
  geotiff_value=0
else
  geotiff_value=1+round((254/20000)*hdf_value) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: cm; range: 0-20cm):

```
if geotiff_value>0
  hdf_value=(geotiff_value-1)*20000/254 //scale 1-255 to 0-20000
  parameter_value=hdf_value*0.001 //apply scaling and offset factors as specified in the HDF SDS
else //geotiff_value=0
  hdf_value=-9999
  parameter_value=NO_RETRIEVAL
end
```

5. MODIS Cloudmask

HDF SDS: Cloud_Mask (generated by IMAPP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
Retrieve bit 0 from byte_1 in HDF SDS
if (bit0 = 0)
  geotiff_value=0
else
  Retrieve bits 2 and 1 (bit21) from byte_1 in HDF SDS
  if (bit21=00) //Cloudy
    geotiff_value=1
  elseif (bit21=01) //Uncertain
    geotiff_value=2
  elseif (bit21=10) //Probably Clear
    geotiff_value=3
  elseif (bit21=11) //Clear
    geotiff_value=4
  endif
endif
```

Pseudo-code to convert GeoTIFF values to actual parameter flags (units: dimensionless):

```
if (geotiff_value=1)
  parameterflag=CLOUDY
elseif (geotiff_value=2)
  parameterflag=UNCERTAIN
elseif (geotiff_value=3)
  parameterflag=PROBABLY_CLEAR
elseif (geotiff_value=4)
  parameterflag=CLEAR
elseif (geotiff_value=0)
  parameterflag=NO_RETRIEVAL
endif
```

6. MODIS NDVI/EVI

HDF SDS: NDVI/EVI (generated by NDVIEVI_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -1000 to 10000 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999) are set to 0.

```

if hdf_value = -999
  geotiff_value=0
elseif pixel has CLOUD or WATER (identified using Active Fire HDF product)
  geotiff_value=0
else
  geotiff_value=1+round((254/11000)*(hdf_value+1000)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: -0.1 to 1.0):

```

if geotiff_value>0
  hdf_value=[(geotiff_value-1)*11000/254]-1000 //scale 1-255 to -1000 to 10000
  parameter_value=hdf_value*0.0001 //apply scaling and offset factors
else //geotiff_value=0
  hdf_value=-999
  parameter_value=NO_RETRIEVAL
end

```

7. MODIS FIRE

HDF SDS: fire mask (generated by MOD14_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
geotiff_value=hdf_value
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter flags (units: dimensionless):

```
hdf_value=geotiff_value
(Flag interpretation: 0- missing, 1- not processed, 2- not processed, 3- water, 4- cloud, 5-non-
fire, 6- unknown, 7- low confidence fire, 8- nominal confidence fire, 9- high confidence fire)
```

8. MODIS Land Surface Temperature

HDF SDS: LST (generated by MODLST_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 2300-3400 (equivalent to 230K-340K) are scaled linearly to 1-255 in GeoTIFF

output. Fill_Values (0) are set to 0.

```
if hdf_value = 0
  geotiff_value=0
elseif (pixel has CLOUD or WATER) (identified using Active Fire HDF product)
  geotiff_value=0
else
  geotiff_value=1+round((254/1100)*(hdf_value-2300)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: K; range: 230K to 340K):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*1100/254]+2300 //scale 1-255 to 2300-3400
  parameter_value=hdf_value*0.1 //apply scaling and offset factors
else //geotiff_value=0
  hdf_value=0
  parameter_value=NO_RETRIEVAL
end
```

9. MODIS Sea Surface Temperature

HDF SDS: sst (generated by L2GEN_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -400 to 9000 (equivalent to -2°C to 45°C) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-32767) are set to 0.

```

if hdf_value = -32767
  geotiff_value=0
elseif qual_sst>=3 (identified using qual_sst sds)
  geotiff_value=0
else
  geotiff_value=1+round((254/9400)*(hdf_value+400)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: °C; range: -2°C to 45°C):

```

if geotiff_value>0
  hdf_value=[(geotiff_value-1)*9400/254]-400 //scale 1-255 to -400 to 9400
  parameter_value=hdf_value*0.005 //apply scaling and offset factors as specified in the HDF
  SDS
else //geotiff_value=0
  hdf_value=-32767
  parameter_value=NO_RETRIEVAL
end

```

10. MODIS Chlorophyll-a Concentration

HDF SDS: chlor_a (generated by L2GEN_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```

if(hdf_value=-32767 or l2flag = Chl_warn or l2flag=Chl_fail)
  geotiff_value=0
else
  if(hdf_value<0.01)
    hdf_value=0.01
  endif
  if(hdf_value>100)
    hdf_value=100
  endif
  geotiff_value= round(128+(63.5*(log10(hdf_value)))
end

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: mg/m³; range: 0.01 to 100):

```
if (geotiff_value=0)
  hdf_value=-1
  parameter_value=NO_RETRIEVAL
else
  hdf_value=10^[(geotiff_value-128)/63.5]
  parameter_value=hdf_value
end
```

11. MODIS True Color Images

MODIS corrected reflectances in bands 1, 4 and 3 generated by CREFL_SPA are used to create the CREFL true color images. Similarly surface reflectances in bands 1, 4 and 3 generated by MOD09_SPA were used to create MOD09 true color images. The scalings used on the red, green and blue bands to create aesthetically pleasing true color images were inspired by Gumley, Descloitres and Schmaltz (2007), "Creating Reprojected True Color MODIS Images: A Tutorial." These standard true color images are available in different resolutions and, optionally, with fire pixel overlays (see Table 2).

12. MODIS True Color Images with Fire Pixel Overlays

MODIS True Color Images (as described above) are enhanced with fire pixels.

13. VIIRS Top of Atmosphere True Color

VIIRS Top of Atmosphere (TOA) reflectances in bands M5, M4 and M3 generated by VIIRS-SDR are used to create the VIIRS TOA true color images. The scalings used were inspired by Gumley, Descloitres and Schmaltz (2007), "Creating Reprojected True Color MODIS Images: A Tutorial."

14. VIIRS CVIIRS True Color

VIIRS corrected reflectances in bands M5 (or I1), M4 and M3 generated by CVIIRS_SPA are used to create the VIIRS true color images. The scalings used were inspired by Gumley, Descloitres and Schmaltz (2007), "Creating Reprojected True Color MODIS Images: A Tutorial."

15. VIIRS CVIIRS True Color with Fire Pixel Overlays

VIIRS true color images generated from CVIIRS (as described above) are enhanced with fire pixel overlays.

16. VIIRS M12 Brightness Temperature

HDF5 SDS: /All_Data/VIIRS-M12-SDR_All/BrightnessTemperature (in SVM12 product generated by VIIRS-SDR_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 14694-54407 (equivalent to 240K-340K) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (65528 to 65535) are set to 0.

```
if hdf_value >= 65528 and hdf_value <= 65535
  geotiff_value=0
else
  geotiff_value=1+round((254/39713)*(hdf_value-14694)) //scale from 1 to 255
  if(geotiff_value < 1)
    geotiff_value=1
  endif
  if(geotiff_value > 255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: K; range: 240K to 340K):

```
if geotiff_value > 0
  hdf_value=[(geotiff_value-1)*39713/254]+14694 //scale 1-255 to 14694 to 54407
  parameter_value=hdf_value*0.0002518046+203.0 //apply scaling and offset factors as
  specified in the HDF SDS
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```

17. VIIRS DNB Radiance (Day time)

HDF5 SDS: /All_Data/VIIRS-DNB-SDR_All/Radiance (in SVDNB product generated by VIIRS-SDR_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0-0.01 (W/(cm² sr)) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-

999.9 to -999.0) are set to 0.

```
if hdf_value >= -999.9 and hdf_value <=-999.0
  geotiff_value=0
else
  geotiff_value=1+round((254/0.01)*(hdf_value)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: W/(cm² sr); range: 0 to 0.01):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*0.01/254] //scale 1-255 to 0 to 0.01
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```

18. VIIRS DNB Radiance (Night time)

HDF5 SDS: /All_Data/VIIRS-DNB-SDR_All/Radiance (in SVDNB product generated by VIIRS-SDR_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0 to 1E-7 (W/(cm² sr)) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999.9 to -999.0) are set to 0.

```
if hdf_value >= -999.9 and hdf_value <=-999.0
  geotiff_value=0
else
  geotiff_value=1+round((254/0.0000001)*(hdf_value)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: W/(cm² sr); range: 0 to 0.0000001):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)* 0.0000001/254] //scale 1-255 to 0 to 0.0000001
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```

19. VIIRS Imagery M Band False Color

VIIRS imagery EDRs for bands M1, M4 and M9 generated by the GTMImagery_SPA are used to create the VIIRS false color images.

20. VIIRS Imagery I Band False Color

VIIRS imagery EDRs for bands I1, I2 and I3 generated by the GTMImagery_SPA are used to create the VIIRS false color images.

21. VIIRS Imagery NCC Albedo

HDF5 SDS: /All_Data/VIIRS-NCC-EDR_All/Albedo (in VNCCO product generated by GTMImagery_SPA).

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0-13105 (equivalent to 0 - 1) are scaled linearly to 1-255 in GeoTIFF output.

Fill_Values (65528 to 65535) are set to 0.

```
if hdf_value >= 65528 and hdf_value <= 65535
  geotiff_value=0
else
  geotiff_value=1+round((254/13105)*(hdf_value)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (unitless, range: 0 to 1):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*13105/254] //scale 1-255 to 14694 to 54407
  parameter_value=hdf_value*0.00007630442 //apply scaling and offset factors as specified in
the HDF SDS
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```

22. VIIRS-AF Fire Mask

HDF SDS: fire mask (generated by VIIRSAF_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
geotiff_value=hdf_value
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter flags (units: dimensionless):

```
hdf_value=geotiff_value
(Flag interpretation: 0- missing, 1- not processed, 2- not processed, 3- water, 4- cloud, 5-non-
fire, 6- unknown, 7- low confidence fire, 8- nominal confidence fire, 9- high confidence fire)
```

23. VIIRS AVAFO Fire Mask

HDF SDS: /All_Data/VIIRS-AF-EDR_All/fireMask (generated by ActiveFires_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
geotiff_value=hdf_value
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter flags (units: dimensionless):

```
hdf_value=geotiff_value  
(Flag interpretation: 0- missing, 1- not processed, 2- not processed, 3- water, 4- cloud, 5-non-  
fire, 6- unknown, 7- low confidence fire, 8- nominal confidence fire, 9- high confidence fire)
```

24. VIIRS Cloud Mask

HDF SDS: /All_Data/VIIRS-CM-IP_All/QF1_VIIRSCMIP (from IICMP product generated by CLOUDMASK_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
Retrieve bit 0 and 1 (bit01) in HDF SDS  
if (bit01 = 0)  
    geotiff_value=0  
else  
    Retrieve bits 2 and 3 (bit23) in HDF SDS  
    if (bit23=00) //Clear  
        geotiff_value=1  
    elseif (bit21=01) //Probably Clear  
        geotiff_value=2  
    elseif (bit21=10) //Probably Cloudy  
        geotiff_value=3  
    elseif (bit21=11) //Confident Cloudy  
        geotiff_value=4  
    endif  
endif
```

Pseudo-code to convert GeoTIFF values to actual parameter flags (units: dimensionless):

```
if (geotiff_value=1)  
    parameterflag= CLEAR  
elseif (geotiff_value=2)  
    parameterflag= PROBABLY_CLEAR  
elseif (geotiff_value=3)  
    parameterflag=PROBABLY_CLOUDY  
elseif (geotiff_value=4)  
    parameterflag=CONFIDENT_CLOUDY  
elseif (geotiff_value=0)  
    parameterflag=NO_RETRIEVAL  
endif
```

25. VIIRS Cloud Phase

HDF SDS: /All_Data/VIIRS-CM-IP_All/QF6_VIIRSCMIP (from IICMP product generated by CLOUDMASK_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
Retrieve bit 0,1 and 2 (bit012) in HDF SDS
if (bit012 = 000)
  geotiff_value=0
else if (bit012=001) //Clear
  geotiff_value=1
elseif (bit012=010) //Partly Cloudy
  geotiff_value=2
elseif (bit012=011) //Water Cloud
  geotiff_value=3
elseif (bit012=100) //Supercooled Water/Moxed
  geotiff_value=4
elseif (bit012=101) //Opaque Ice Cloud
  geotiff_value=5
elseif (bit012=110) //Cirrus Cloud
  geotiff_value=6
elseif (bit012=111) //Cloud Overlap
  geotiff_value=7
endif
```

Pseudo-code to convert GeoTIFF values to actual parameter flags (units: dimensionless):

```
if (geotiff_value=1)
  parameterflag= CLEAR
elseif (geotiff_value=2)
  parameterflag= PARTLY_CLOUDY
elseif (geotiff_value=3)
  parameterflag=WATER_CLOUD
elseif (geotiff_value=4)
  parameterflag=MIXED_CLOUD
elseif (geotiff_value=5)
  parameterflag=ICE_CLOUD
elseif (geotiff_value=6)
  parameterflag=CIRRUS_CLOUD
elseif (geotiff_value=7)
  parameterflag=CLOUD_OVERLAP
elseif (geotiff_value=0)
  parameterflag=NO_RETRIEVAL
endif
```

26. VIIRS Aerosol Optical Thickness

HDF5 SDS: /All_Data/VIIRS-Aeros-Opt-Thick-IP_All/faot550 (in IVAOT product generated by Aerosol_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0.0 to 5.0 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999) in SDS are set to 0 in geotiff.

```
if hdf_value = -999
  geotiff_value=0
else
  geotiff_value=1+round((254/5.0)*hdf_value) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: 0-5):

```
if geotiff_value>0
  hdf_value=(geotiff_value-1)*5.0/254 //scale 1-255 to 0-5.0
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=-999
  parameter_value=NO_RETRIEVAL
end
```

26. VIIRS Aerosol Particle Size

HDF5 SDS: /All_Data/VIIRS-Aeros-Opt-Thick-IP_All/angexp (in IVAOT product generated by Aerosol_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0.0 to 3.0 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999) in SDS are set to 0 in geotiff.

```
if hdf_value = -999
  geotiff_value=0
else
  geotiff_value=1+round((254/3.0)*hdf_value) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
```

```

        geotiff_value=255
    endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: 0-3.0):

```

if geotiff_value>0
    hdf_value=(geotiff_value-1)*3.0/254 //scale 1-255 to 0-3.0
    parameter_value=hdf_value
else //geotiff_value=0
    hdf_value=-999
    parameter_value=NO_RETRIEVAL
end

```

27. VIIRS Suspended Matter

HDF SDS: /All_Data/VIIRS-SusMat-EDR_All/SuspendedMatterType (from VSUMO product generated by AEROSOL_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```

if hdf_value >= 249 and hdf_value<=255
    geotiff_value=0
else
    geotiff_value=1+hdf_value
    if(geotiff_value<1)
        geotiff_value=1
    endif
    if(geotiff_value>255)
        geotiff_value=255
    endif
endif

```

Pseudo-code to convert GeoTIFF values to actual parameter flags (units: dimensionless):

```

if (geotiff_value>0)
    hdf_value=geotiff_value-1
elseif (geotiff_value=0)
    hdf_value=NO_VALUE
endif

```

28. VIIRS Cloud Top Temperature

HDF5 SDS: /All_Data/VIIRS-INWCTT-IP_All/ctt (in IVIWT product generated by COP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 180.0 to 300.0 (equivalent to 180K-300K) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999) are set to 0.

```
if hdf_value >= -1000 and hdf_value <=-999
  geotiff_value=0
else
  geotiff_value=1+round((254/120)*(hdf_value-180)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: K; range: 180K to 300K):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*120/254]+180 //scale 1-255 to 180 to 300
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```

29. VIIRS Cloud Optical Thickness

HDF5 SDS: /All_Data/VIIRS-Cd-Opt-Prop-IP_All/cot (in IVCOP product generated by COP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0.0 to 40.0 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999) in SDS are set to 0 in geotiff.

```
if hdf_value >= -1000 and hdf_value <=-999
  geotiff_value=0
else
  geotiff_value=1+round((254/40.0)*hdf_value) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

```
endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: 0-40.0):

```
if geotiff_value>0
  hdf_value=(geotiff_value-1)*40.0/254 //scale 1-255 to 0-40.0
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=-999
  parameter_value=NO_RETRIEVAL
end
```

30. VIIRS Effective Particle Size

HDF5 SDS: /All_Data/VIIRS-Cd-Opt-Prop-IP_All/eps (in IVCOP product generated by COP_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 0.0 to 40.0 (microns) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999) in SDS are set to 0 in geotiff.

```
if hdf_value >= -1000 and hdf_value<=-999
  geotiff_value=0
else
  geotiff_value=1+round((254/40.0)*hdf_value) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: microns; range: 0-40.0):

```
if geotiff_value>0
  hdf_value=(geotiff_value-1)*40.0/254 //scale 1-255 to 0-40.0
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=-999
  parameter_value=NO_RETRIEVAL
end
```

31. VIIRS Snow Binary Map

HDF SDS: /All_Data/VIIRS-SCD-BINARY-SNOW-MAP-EDR_All/SnowCoverBinaryMap (from VSCMO product generated by SNOWCOVER_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
if hdf_value >= 249 and hdf_value<=255
  geotiff_value=0
elseif QF1_VIIRSSCDBINARYSNOWMAPEDR is 'BAD' or 'NO-RETRIEVAL'
  geotiff_value=0
else
  geotiff_value=1+hdf_value
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to actual parameter flags (units: dimensionless):

```
if (geotiff_value>0)
  hdf_value=geotiff_value-1
elseif (geotiff_value=0)
  hdf_value=NO_VALUE
endif
```

32. VIIRS NDVI

HDF SDS: /All_Data/VIIRS-VI_EDR_All/TOA_NDVI (generated by VegIndex_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -29428 to 65527 (equivalent to NDVI values -0.1 to 1.0) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (65528 to 65535) are set to 0.

```

if hdf_value >= 65528 and hdf_value<=65535
  geotiff_value=0
elseif pixel has CLOUD, WATER or FILL (identified using AVAFO HDF product)
  geotiff_value=0
else
  geotiff_value=1+round((254/36099)*(hdf_value-29428)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: -0.1 to 1.0):

```

if geotiff_value>0
  hdf_value=[(geotiff_value-1)*36099/254]+29428 //scale 1-255 to 29428-65527
  parameter_value=hdf_value*0.00003052177-1 //apply scaling and offset factors
else //geotiff_value=0
  hdf_value=65535
  parameter_value=NO_RETRIEVAL
end

```

33. VIIRS EVI

HDF SDS: /All_Data/VIIRS-VI_EDR_All/TOC_EVI (generated by VegIndex_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -11795 to 26211 (equivalent to EVI values -0.1 to 1.0) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (65528 to 65535) are set to 0.

```

if hdf_value >= 65528 and hdf_value<=65535
  geotiff_value=0
elseif pixel has CLOUD, WATER or FILL (identified using AVAFO HDF product)
  geotiff_value=0
else
  geotiff_value=1+round((254/14416)*(hdf_value-11795)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: -0.1 to 1.0):

```

if geotiff_value>0
  hdf_value=[(geotiff_value-1)*14416/254]+11795 //scale 1-255 to 11795 to 26211
  parameter_value=hdf_value*0.00007630442-1 //apply scaling and offset factors
else //geotiff_value=0
  hdf_value=65535
  parameter_value=NO_RETRIEVAL
end

```

34. VIIRS LST

HDF SDS: /All_Data/VIIRS-LST-EDR_All/LandSurfaceTemperature (generated by LST_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -18385 to 61598 (equivalent to LST values 230K to 340K) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (65528 to 65535) are set to 0.

```

if hdf_value >= 65528 and hdf_value<=65535
  geotiff_value=0
elseif pixel has CLOUD, WATER or FILL (identified using AVAFO HDF product)
  geotiff_value=0
else
  geotiff_value=1+round((254/43213)*(hdf_value-18385)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: Kelvin; range: 230K to 340K):

```

if geotiff_value>0
  hdf_value=[(geotiff_value-1)*43213/254]+18385 //scale 1-255 to 18385 to 61598
  parameter_value=hdf_value*0.0025455155+183.2 //apply scaling and offset factors
else //geotiff_value=0
  hdf_value=65535
  parameter_value=NO_RETRIEVAL
end

```

35. VIIRS L2GEN SST

HDF SDS: sst (generated by L2GEN_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -400 to 9400 (equivalent to -2°C to 45°C) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-32767) are set to 0.

```
if hdf_value = -32767
  geotiff_value=0
elseif qual_sst>=3 (identified using qual_sst sds)
  geotiff_value=0
else
  geotiff_value=1+round((254/9400)*(hdf_value+400)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: °C; range: -2°C to 45°C):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*9400/254]-400 //scale 1-255 to -400 to 9400
  parameter_value=hdf_value*0.005 //apply scaling and offset factors as specified in the HDF
  SDS
else //geotiff_value=0
  hdf_value=-32767
  parameter_value=NO_RETRIEVAL
end
```

36. VIIRS L2GEN Chlorophyll-a Concentration

HDF SDS: chlor_a (generated by L2GEN_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values:

```
if(hdf_value=-32767 or l2flag = Chl_warn or l2flag=Chl_fail)
  geotiff_value=0
else
  if(hdf_value<0.01)
    hdf_value=0.01
  endif
  if(hdf_value>100)
    hdf_value=100
  endif
  geotiff_value= round(128+(63.5*(log10(hdf_value)))
end
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter

values (units: mg/m³; range: 0.01 to 100):

```
if (geotiff_value=0)
  hdf_value=-1
  parameter_value=NO_RETRIEVAL
else
  hdf_value=10^[(geotiff_value-128)/63.5]
  parameter_value=hdf_value
end
```

37. VIIRS NDVI/EVI (from VIIRS-VI SPA)

HDF SDS: NDVI/EVI (generated by VIIRS-VI_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -1000 to 10000 are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-999) are set to 0.

```
if hdf_value = -999
  geotiff_value=0
elseif pixel has CLOUD or WATER (identified using VAF HDF product from VIIRS-AF_SPA)
  geotiff_value=0
else
  geotiff_value=1+round((254/11000)*(hdf_value+1000)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: -0.1 to 1.0):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*11000/254]-1000 //scale 1-255 to -1000 to 10000
  parameter_value=hdf_value*0.0001 //apply scaling and offset factors
else //geotiff_value=0
  hdf_value=-999
  parameter_value=NO_RETRIEVAL
end
```

38. VIIRS Land Surface Temperature (from VIIRS-LST)

HDF SDS: LST (generated by VIIRS-LST_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from 2300-3400 (equivalent to 230K-340K) are scaled linearly to 1-255 in GeoTIFF

output. Fill_Values (0) are set to 0.

```
if hdf_value = 0
  geotiff_value=0
elseif (pixel has CLOUD or WATER) (identified using VAF HDF product from VIIRS-AF_SPA)
  geotiff_value=0
else
  geotiff_value=1+round((254/1100)*(hdf_value-2300)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: K; range: 230K to 340K):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*1100/254]+2300 //scale 1-255 to 2300-3400
  parameter_value=hdf_value*0.1 //apply scaling and offset factors
else //geotiff_value=0
  hdf_value=0
  parameter_value=NO_RETRIEVAL
end
```

39. OMPS Ultraviolet Aerosol Index

HDF5 SDS: /SCIENCE_DATA/UVAerosolIndex (in OMPS Total Column Total Ozone product generated by OMPSnadir_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -2.0 to 4.0 (equivalent to aerosol index of -2.0 to 4.0) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-9999 to -3000) are set to 0.

```
if hdf_value >= -9999 and hdf_value <= -3000
  geotiff_value=0
else
  geotiff_value=1+round((254/6.0)*(hdf_value+2.0)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless; range: -2.0 to 4.0):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*6.0/254] -2.0 //scale 1-255 to -2.0 to 4.0
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```

40. OMPS Total Column Ozone

HDF5 SDS: /SCIENCE_DATA/ColumnAmountO3 (in OMPS Total Column Total Ozone product generated by OMPSnadir_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -100.0 to 500.0 (equivalent to Total Ozone of 100 D.U. to 500.0 D.U) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-9999 to 50) are set to 0.

```
if hdf_value >= -9999 and hdf_value <= 50
  geotiff_value=0
else
  geotiff_value=1+round((254/400)*(hdf_value-100)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif
```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: D.U. ; range: 100.0 to 500.0):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*400.0/254] +100.0 //scale 1-255 to 100-500
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```

41. OMPS Reflectivity at 331nm

HDF5 SDS: /SCIENCE_DATA/Reflectivity331 (Total Column Total Ozone product generated by OMPSnadir_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -0.0 to 1.0 (equivalent to Reflectivity of 0.0 to 1.0) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-9999 to -1) are set to 0.

```

if hdf_value >= -9999 and hdf_value <=-1.0
  geotiff_value=0
else
  geotiff_value=1+round((254/1.0)*(hdf_value)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: dimensionless ; range: 0.0 to 1.0):

```

if geotiff_value>0
  hdf_value=[(geotiff_value-1)*1.0/254] //scale 1-255 to 0.0-1.0
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end

```

42. OMPS Total Column SO₂

HDF5 SDS /HDFEOS/SWATHS/OMPS Column Amount SO₂/Data Fields/ColumnAmountSO₂_TRM_(in OMPS SO₂ product generated by OMPSnadir_SPA)

Scaling used to convert HDF-SDS values to GeoTIFF values: HDF SDS data from -0.0 to 3.0 (equivalent to Total Ozone of 0.0 D.U. to 3.0 D.U) are scaled linearly to 1-255 in GeoTIFF output. Fill_Values (-9999 to -10) are set to 0.

```

if hdf_value >= -9999 and hdf_value <=-10
  geotiff_value=0
else
  geotiff_value=1+round((254/3.0)*(hdf_value)) //scale from 1 to 255
  if(geotiff_value<1)
    geotiff_value=1
  endif
  if(geotiff_value>255)
    geotiff_value=255
  endif
endif

```

Pseudo-code to convert GeoTIFF values to HDF SDS values/actual parameter values (units: D.U. ; range: 0.0 to 3.0):

```
if geotiff_value>0
  hdf_value=[(geotiff_value-1)*3.0/254] //scale 1-255 to 0.0-3.0
  parameter_value=hdf_value
else //geotiff_value=0
  hdf_value=FILL_VALUE
  parameter_value=NO_RETRIEVAL
end
```